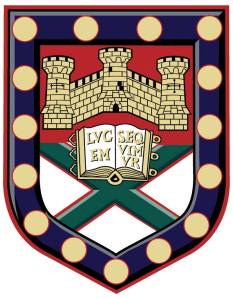
Cross Entropy-based Analysis of Spacecraft Control Systems



Anusha Pradeep Mujumdar

University of Exeter

Doctor of Philosophy in Mathematics

In October 2016

I certify that all material in this thesis which is not my own work has been identified and that no material has previously been submitted and approved for the award of a degree by this or any other University.

Signature: _____

This thesis is dedicated to

my late mother for preparing me for the journey,

my father for showing me the way, and

Srikanth, for walking every step with me.

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Dr. Prathyush Menon, for his guidance and support throughout my doctoral studies. I am indebted to him for the intellectual freedom he gave me, at the same time providing me many opportunities to work on interesting research projects, all of which have led to my academic development. I am thankful to him for several insightful discussions, and for sharing his rich knowledge and experience in aerospace control systems. I feel privileged to have worked with and learned from him. I thank Prof. Christopher Edwards, my second supervisor, for his valuable help and guidance, and for setting an example for me to aspire to.

I am thankful to Prof. Declan Bates for engaging discussions during the first two years of my studies, and his continued encouragement and support since. I am indebted to Prof. Chris Ferro for being a gentle, patient mentor whose counsel was invaluable to me at crucial times during my PhD. I am fortunate to have worked with Dr. Samir Bennani, Hans Strauch and Prof. Debasish Ghose. It is an honour to have been mentored by such world-renowned experts in their fields, and the inspiration I have gained from interacting with them will remain with me throughout my career. My thanks also go to Nuno Paulino, Luigi Strippoli, Dr. Finn Ankersen and Christina Jetzschmann for enriching collaborations. I am thankful to Dr. Premila Manohar and Dr. Radhakant Padhi, who encouraged me to pursue

doctoral education. I gratefully acknowledge that my PhD research work was made possible by the financial support extended by the European Space Agency. I am also thankful to the Zonta foundation for awarding me the Amelia Earhart fellowship.

I am thankful to my lab colleagues Atul, Siyi, Chiara and Bharani for an enjoyable working environment. I would also like to express my gratitude to the postgraduate administration team, particularly Liz Roberts, for their support and assistance. I am thankful to my friends, Laurianne, Stephanie, Sneha, Varun, Nick, Helen, Darren, Akkas, Emma, Eugenia, Lineke, Marta and Jack, for making the time I spent in Exeter beautiful.

I thank my parents for their unconditional love, guidance and support. I am grateful to my sister Neha for being a strong, inspiring example, and for reading my thesis and providing valuable feedback. My husband Srikanth has been an unending source of support and compassion, for which I am deeply grateful. I would like to thank my parents-in-law for their patience, understanding and support while I was writing my thesis. I am grateful to Vinutha, Winfred, Srinidhi, Shreya, Madhura, Puneeth, Rohit, Arjun, Manasa, Shashiprakash and Neha Satak for their love and friendship.

Abstract

Space missions increasingly require sophisticated guidance, navigation and control algorithms, the development of which is reliant on verification and validation ($V\&V$) techniques to ensure mission safety and success. A crucial element of $V\&V$ is the assessment of control system robust performance in the presence of uncertainty. In addition to estimating average performance under uncertainty, it is critical to determine the worst case performance. Industrial $V\&V$ approaches typically employ $\boldsymbol{\mu}$-analysis in the early control design stages, and Monte Carlo simulations on high-fidelity full engineering simulators at advanced stages of the design cycle. While highly capable, such techniques present a critical gap between pessimistic worst case estimates found using analytical methods, and the optimistic outlook often presented by Monte Carlo runs. Conservative worst case estimates are problematic because they can demand a controller redesign procedure, which is not justified if the poor performance is unlikely to occur.

Gaining insight into the probability associated with the worst case performance is valuable in bridging this gap. It should be noted that due to the complexity of industrial-scale systems, $V\&V$ techniques are required to be capable of efficiently analysing non-linear models in the presence of significant uncertainty. As well, they must be computationally tractable. It is desirable that such techniques demand little engineering effort before each

analysis, to be applied widely in industrial systems.

Motivated by these factors, this thesis proposes and develops an efficient algorithm, based on the cross entropy simulation method. The proposed algorithm efficiently estimates the probabilities associated with various performance levels, from nominal performance up to degraded performance values, resulting in a curve of probabilities associated with various performance values. Such a curve is termed the probability profile of performance (PPoP), and is introduced as a tool that offers insight into a control system's performance, principally the probability associated with the worst case performance.

The cross entropy-based robust performance analysis is implemented here on various industrial systems in European Space Agency-funded research projects. The implementation on autonomous rendezvous and docking models for the Mars Sample Return mission constitutes the core of the thesis. The proposed technique is implemented on high-fidelity models of the Vega launcher, as well as on a generic long coasting launcher upper stage.

In summary, this thesis (a) develops an algorithm based on the cross entropy simulation method to estimate the probability associated with the worst case, (b) proposes the cross entropy-based PPoP tool to gain insight into system performance, (c) presents results of the robust performance analysis of three space industry systems using the proposed technique in conjunction with existing methods, and (d) proposes an integrated template for conducting robust performance analysis of linearised aerospace systems.

Contents

List of Figures

List of Tables

Nomenclature and Abbreviations

δ	Uncertain parameter vector
$\boldsymbol{\mu}$	Structured singular value
μ, σ	Mean and standard deviations respectively of a distribution
$\bar{S}_v$	Maximum singular value
γ	Performance level
X, U, Y	State, input and output vectors of a system
A, B, C, D	State-space model matrices
J	Performance function
$V\&V$	Verification and validation
WCA	Worst case analysis
LFT	Linear fractional transformation
MC	Monte Carlo
GNC	Guidance, navigation and control
iGNC	Integrated GNC of autonomous rendezvous
FES	Full engineering simulator
RP	Robust performance
CE	Cross entropy
PPoP	Probability profile of performance

Chapter 1

Introduction

Space flight has been a significant human achievement in the last century; space missions have led to paradigmatic advances in science and technology. Space mission failures, per contra, can be catastrophic, leading to loss of mission, money or even human life. For example, NASA's climate orbiter, which was to study the Martian atmosphere and relay radio signals from two surface probes, was destroyed during orbital insertion. Subsequent analysis found that the reason for this failure was thruster software that carried out computations in Imperial units, while the rest of the computation was in SI units. As a result, the probe was guided to Mars' upper atmosphere, which led to its disintegration [1]. Another failure of relevance was the Demonstration for Autonomous Rendezvous Technology (DART) mission, which was to perform proximity manoeuvres with a target spacecraft already in orbit. However, inaccurate estimation of distance and speed resulted in the spacecraft colliding with its target. It failed to meet all 14 of its mission objectives, and was declared a Type A mishap[1] due to the mission incurring a loss of more than 1 million US dollars [2]. More recently, the CRS-7 Dragon mission [3], which was to resupply the International Space Station in June 2015, failed due to an overpressure event and eventual explosion in the Falcon-9 launcher [4].

[1] This level of mishap requires the most rigorous investigation according to NASA's protocols

A recent paper [5] investigates the contribution of software in catastrophic accidents including several aerospace accidents and mishaps. As part of their recommendations, the authors argue that testing for unsafe behaviour is as important as testing for expected performance. They also cite the DART mishap and point out that safety-critical software should be able to recognise and be unresponsive to inaccurate data.

Unfortunately, there exist several instances of partial or total mission failure, some of which resulted in deaths. A study [6] of 25 years of on-orbit failures from 1980 onwards observes that such failures occur despite the rigorous testing processes in place, and that each space mission is challenging in unique ways. The study finds that 59% of all on-orbit spacecraft failures are due to the Attitude and Orbit Control System (AOCS) and power subsystems. The AOCS is responsible for the guidance, navigation and control (GNC) of the spacecraft, and for meeting mission objectives in the face of disturbances and uncertainties arising during flight. In [6] the author notes that 41% of all on-orbit failures occur within a year of reaching orbit, which suggests "insufficient testing and inadequate modelling of the spacecraft and its environment." It is therefore clear that the critical problem of spacecraft safety is one that is tied to the development and analysis of safe, reliable and high-performance GNC systems.

In the last few years, space aspirations have grown to include new directions, such as space tourism, asteroid mining, space structures maintenance, satellite self-assembly in orbit and human settlements on Mars. Private space companies such as SpaceX, Blue Origin and MarsOne are looking to rapidly develop novel space technologies, while national and international space agencies have also set ambitious targets [7]. Safety, an already critical factor in space flight, takes on even greater importance when such complex missions are partially or wholly autonomous. The development of autonomous

flight control laws is a complicated and highly multi-disciplinary task, making their clearance process challenging.

Sophisticated GNC algorithms are key to enabling present and future space missions [8]. GNC algorithms can steer space missions to success with precision, leveraging computational resources to achieve ambitious objectives [9]. Such algorithms also save time, engineering effort, costs and eliminate human errors. The process of synthesising GNC algorithms ideally consists of iterations of control system design and analysis until requirements are satisfactorily met, as shown in Figure 1.1 — this figure is adapted from [10], and is similar to the so-called V model in the broad software testing community.

Synthesis of GNC algorithms [1] in practical space systems is often carried out on a nominal, linearised system model [11]. In practice, however, spacecraft dynamical systems are large high-fidelity systems with non-linearities and several uncertainties, as shown in Figure 2.3. Uncertainty in aerospace systems may exist due to modelling inadequacies (such as roll coupling, fuel sloshing and flexible modes) or external disturbances (such as wind velocities and disturbance torques).

Therefore, the designed control algorithms need to be assessed for safety and performance under realistic operating conditions — via a process known as Verification and Validation. Assessing robust performance in the presence of uncertainty, is key in aerospace verification and validation. The prevalent methodology for robust performance analysis in industrial spacecraft systems is to conduct a μ-analysis [12] on a simplified model in the early design stage, followed by large-scale Monte Carlo runs [13] on high-fidelity models. However, μ-analysis methods are known to provide rather pes-

[1] referred to as control synthesis in the sequel, or simply synthesis. Likewise, the umbrella term *control* refers to GNC algorithms.

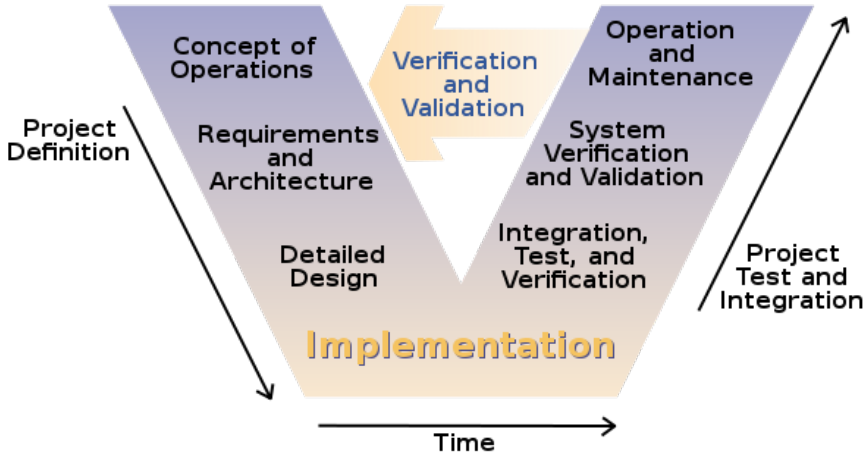


Figure 1.1: Control system design cycle including iterations of design and testing Source: NASA.

simistic values of the worst case performance, meaning that the worst case performance predicted by it may be catastrophic. Such performances may have low likelihoods of occurrence. This issue of conservatism has been researched [14], and it is generally accepted that assigning probabilities to worst case performances, i.e. quantifying conservatism, provides a better view of performance. Monte Carlo can provide probabilistic information, and indeed is a good technique to estimate the expected performance and the standard deviation, but large computational burdens are incurred in accurately arriving at extreme values and its probability. This presents a gap between expected performance and the worst case performance that needs to be filled.

This thesis proposes a new technique, based on cross entropy (CE) simulation, to probabilistically assess the performance of spacecraft under the influence of uncertainty. The worst-case performance is expected to have a low likelihood of occurrence, and therefore is interpreted as a rare event. The cross-entropy technique has been established in the literature to be an excellent technique for rare-event probability estimation.

This study considers the robust performance assessment for three space industry models, at varying levels of system complexity, and each challenging in unique ways. The models considered are: the autonomous rendezvous of the Mars Sample Return mission, the Vega launcher in its atmospheric flight phase, and a generic long-coasting launcher upper stage attitude model. The CE-based methodology is implemented on these models, often in conjunction with existing techniques.

This section presented the background and motivation behind the research work presented in this thesis. The contributions of the thesis, and its outline are discussed in the following section.

1.1 Contributions and Outline of the Thesis

The main contributions of this thesis are:

- The cross-entropy simulation technique is developed as a tool to assess the robust performance of controllers, and to determine the probability associated with the worst case performance.
- The probability profile of performance is introduced as a tool to gain insight into the probabilistic characteristics of control system performance.
- An integrated analysis template is proposed to analyse linearised aerospace control systems.
- Detailed linear fractional transformation models are developed for the linearised models of the autonomous rendezvous of the Mars sample return mission.

These contributions will be detailed in the following chapters, which are structured as follows.

Chapter 2 provides an overview of existing robust performance analysis techniques in the literature. The chapter discusses some desirable characteristics of aerospace $V\&V$ techniques, and discusses uncertainty and its representation in aerospace systems. Subsequently, the chapter touches upon commonly used performance measures, and proceeds to survey the robust performance analysis techniques. These techniques are classified as analytical, Monte Carlo based, optimisation-based and probabilistic robustness analysis techniques. Preliminary implementation details of the $\boldsymbol{\mu}$-analysis and the Differential Evolution (DE) optimisation method are presented. The crucial notions of robustness (both in the deterministic and the more recent probabilistic sense) are explained.

Chapter 3 details the cross entropy-based simulation technique, and develops it into a tool for probabilistic analysis of aerospace control systems. Some preliminaries on probability theory are first presented, followed by a detailing on how they fit into the robustness analysis paradigm. The probability profile of performance is then introduced, followed by a discussion of the probabilistic insights into performance provided by it. The proposed technique is then implemented on a simple illustrative example.

In Chapter 4, the Mars Sample Return autonomous rendezvous models are detailed. Linear Fractional Transformation models, crucial for robust analysis using analytical methods, are developed and validated for a set of linearised models, both from scratch and using tools available within Matlab. These linearised models when interconnected form the simplified rendezvous model. A more complex, high-fidelity, non-linear model is briefly described. Both the simplified and the high-fidelity models are to be analysed for robust performance.

Chapter 5 presents the results of robust performance analysis on the rendezvous mod-

els. First, a rigorous early design stage analysis of the simplified model is presented, so as to analyse the critical sensitivity and complementary sensitivity functions of the closed loop system. An optimisation-based analysis and a μ-analysis are implemented to determine the worst case performance and the robust performance margins. In addition, the CE-based technique developed in Chapter 3 is implemented to determine the probability profile of performance. It is found that CE complements the more mature techniques, by confirming the worst case performance, and by assigning a probability to its occurrence.

In Chapter 6, the CE technique is applied in two industrial launch vehicle models. The first is an attitude controller for the spinning motion of a long-coasting upper stage. The implementation is done in conjunction with DE. The unique characteristic of this analysis is that payload configuration parameters are included in the analysis. This analysis is in the early design stage. The second part of the chapter analyses the Vega launch vehicle for robust performance. This is carried out in the late design stage. It is hoped that such an implementation can increase the technological maturity of CE. Chapter 7 draws conclusions from the study, and explains the implications of the work in aerospace safety. Some recommendations for future research are presented.

As seen in the inter-relationship between the chapters, illustrated in Figure 1.2, Chapters 2 and 3 may be considered as background and methods chapters, which provide an important foundation to the thesis. Chapter 4 is purely focussed on the models, useful in the analysis presented in Chapter 5, which forms the heart of the thesis. Chapter 6, which presents further results on two industry models, also uses the methods described in Chapters 2 and 3 . Finally, the conclusions follow from the results presented in Chapters 5 and 6, and implicitly from the background and methods Chapters 2 and 3.

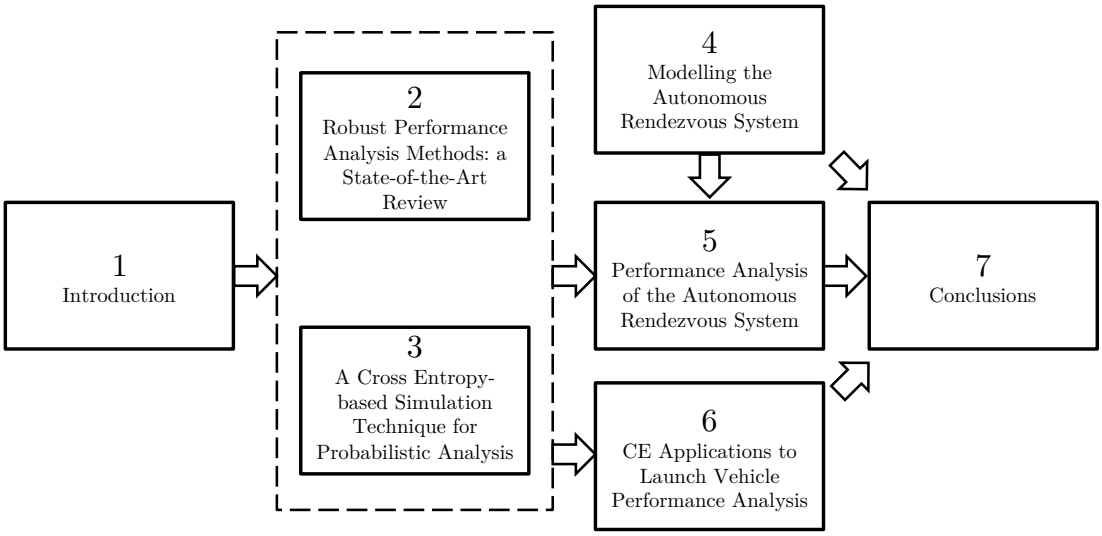


Figure 1.2: Chapters in this thesis and their interrelationships.

Chapter 2

Robust Performance Analysis Methods: a State-of-the-Art Review

The efficient development of advanced aerospace GNC systems is highly dependent on the availability of reliable V&V techniques. $V\&V$ is the step in the design process necessary for assessing and ensuring safety, and for clearing spacecraft missions. One of the greatest challenges in $V\&V$ is accounting for the effects of uncertainty, particularly in the highly unpredictable environment of space. Uncertainty is accounted for in control synthesis using robust control methods [15]. Robust controllers are designed to cope with modelled uncertainties, such that the system's stability and performance remain in close vicinity of their desired levels. However, in practice, robust control design is performed using an assumed viable model, which when plugged in to the high fidelity model often yields undesirable behaviours. In the worst case this could lead to loss of mission. There exists a need, therefore, for assessing and quantifying such undesirable behaviour.

This chapter presents an overview of robust performance analysis methods in the literature, with a focus on their application in aerospace $V\&V$. The desirable qualities of a good $V\&V$ technique are first listed in Section 2.1. Then, the concept of uncertainty is formalised within the robust control context in Section 2.2. Section 2.3 presents measures of performance, and Section 2.4 formalises the robust performance analysis problem. The robust performance techniques in the literature, classified as analytical, Monte Carlo-based, optimisation-based and probabilistic methods, are presented from Section 2.5 onwards. This chapter establishes fundamental concepts (for example, those of robust performance, worst case performance, and probabilistic robust performance) that will be utilised in the rest of the thesis.

2.1 Background

Verification is defined in [16] as: "the process of determining that a model implementation accurately represents the developer's conceptual description of the model and the solution to the model," while validation is "the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model." Within the aerospace industria, the process of $V\&V$ is also known as clearance of flight control laws [14], which involves proving that the designed control laws enable safe flight "under all parameter variability and failure conditions." Such a process may consist of several types of tests, such as robustness analysis, worst-case analysis and analysis of statistical measures such as standard deviation and risk [14], as shown in Figure 2.1[1]. These concepts are discussed in this chapter in some detail. Note that unlike aircraft controller clearance, an accurate physical testing environment cannot be created for space mission $V\&V$. Zero gravity and orbital dynamics must necessarily be mathematically modelled and their effects studied

[1] This figure is sourced from [17]

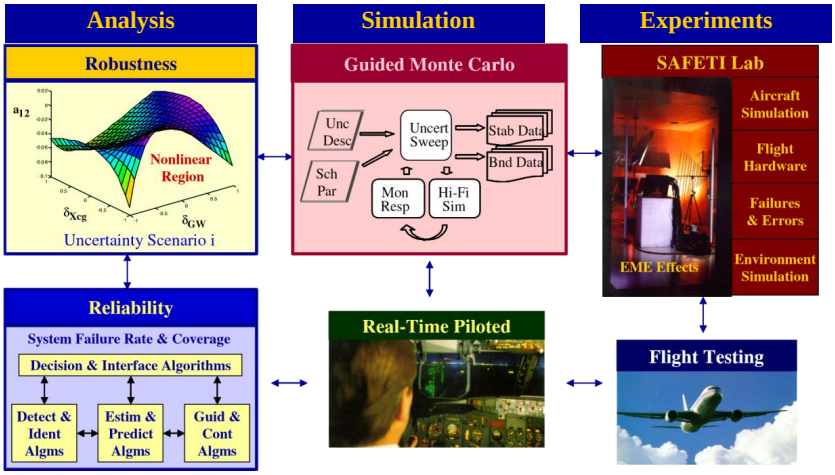


Figure 2.1: $V\&V$ processes for flight clearance. Source: NASA

through simulation. Several physical effects must be accounted for in $V\&V$ for space control systems. A good technique for spacecraft $V\&V$ must ideally meet the following guidelines [11]:

- it should account correctly for parameter uncertainties and unmodelled dynamics (including couplings in position and attitude);
- it should, implicitly or explicitly, consider accurately the control objectives;
- it should consider the effects of fuel sloshing, flexible modes, bending modes and other specific physical issues in the specific spacecraft or launcher;
- sensor errors such as noise level and bias should be considered; and
- it should consider disturbance torques, saturation and other effects in the actuators, i.e., the reaction wheels and thrusters.

In addition, some desirable characteristics of $V\&V$ methods [18] are as below.

- Easily adaptable to various types of systems: it should ideally apply to nonlinear, time-varying systems with various kinds of uncertainty descriptions, time delays, varying initial conditions and other unmodelled effects.

- Easy implementability within existing industry frameworks, with minimal engineering effort before each analysis.
- High maturity of the available analysis tools: they should have demonstrated high efficacy in different applications and, possibly, in other space missions. In other words, the tools should have a high Technology Readiness Level (TRL) [19].
- High accuracy of the technique, with a quantifiable confidence level in the solution.
- Availability of software tools or packages for ease of implementation.
- Applicability to the control synthesis process: solutions obtained should provide indications on tuning and improving the control design [20].

$V\&V$ analysis of the control system is performed at various steps in the design process. Different models of the plant[1] may be used at each step, and consequently the analysis tools involved are different. In general, the industry considers the following phases of the synthesis process for the control law analysis:

- Design stage analysis of the controller: Typically, control laws are designed using low order viable representations of the system. A design stage analysis is generally conducted in a linear model environment with tools that are specific to the synthesis domain. In such an environment, only those models relevant to the control loop are present (without the guidance or navigation loops). At this stage, the performance of the controller is usually validated against requirements expressed in the frequency domain (e.g., stability and performance margins) [21].
- Full Engineering Simulator (FES) analysis: the FES for detailed performance assessment contains high fidelity models. Such models can contain nonlinear dynamics, initial conditions and may be in the time domain. This analysis typically

[1] In this case the plant represents the spacecraft dynamics.

takes place towards the end of the control development cycle. In this phase, the control performances can be evaluated against specific mission requirements, defined in the mission domain (e.g., fulfilment of approach corridor requirements or fuel consumption criteria) [22].The navigation and guidance loops are typically included at this stage of the analysis.

- Control performance assessment in a Processor-in-the-Loop environment: at this stage, the criteria are specified in terms of mission completeness, and the focus is placed on the evaluation of real-time implementation and integrity of the algorithms on a specific hardware platform. Additional performance indicators related to CPU load and algorithm footprint are measured in this phase [9].

At every phase of the design process the interest is in determining both whether the closed-loop control system is stable and whether it fulfils specific performance criteria. Additionally, it is important to assess whether these characteristics are achieved both for the nominal case and in cases where the plant deviates from the nominal case inside the allowed (expected) boundaries. Figure 2.2 shows an illustration of a hypothetical system's response under uncertainty. The black curve shows the nominal model response, i.e., with no uncertainties included. In reality, due to the presence of uncertainty in the plant, the responses lie within the envelope represented by the green dashed curves. The worst case performance is often far worse in the presence of uncertainty, and must be studied carefully.

It must be pointed out that in the wider $V\&V$ literature, considerations of uncertainty are primarily included during validation [16], while verification deals with a conceptual model reminiscent of the nominal model. However, in aerospace systems, uncertainty is of importance in both verification and validation. The concept of robustness is therefore at the core of modern flight control analysis (and indeed, synthesis) to guarantee

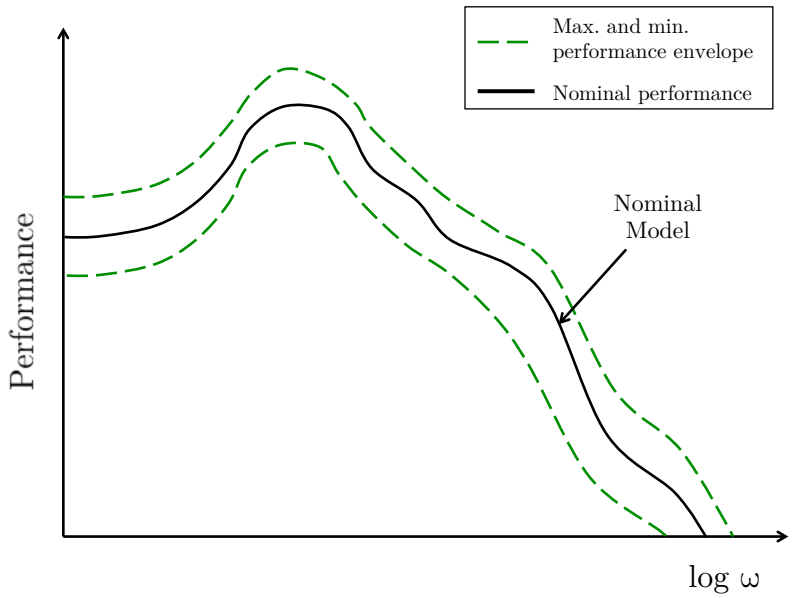


Figure 2.2: Nominal system response and uncertain system response envelopes

both safety and expected performance under all modelled uncertainty [23]. Such requirements are captured by the following properties of a closed-loop control system: Nominal Stability (NS), Robust Stability (RS), Nominal Performance (NP) and Robust Performance (RP).

These notions of robustness are elucidated in [15]. The discussion in this thesis is primarily focused on nominal and robust *performance*, and excludes nominal and robust stability for the following reasons:

- Nominal stability is straightforward to determine for the class of plant dynamics considered in this thesis, and
- Performance degradation occurs long before the onset of instability in most cases, as stated in [24].

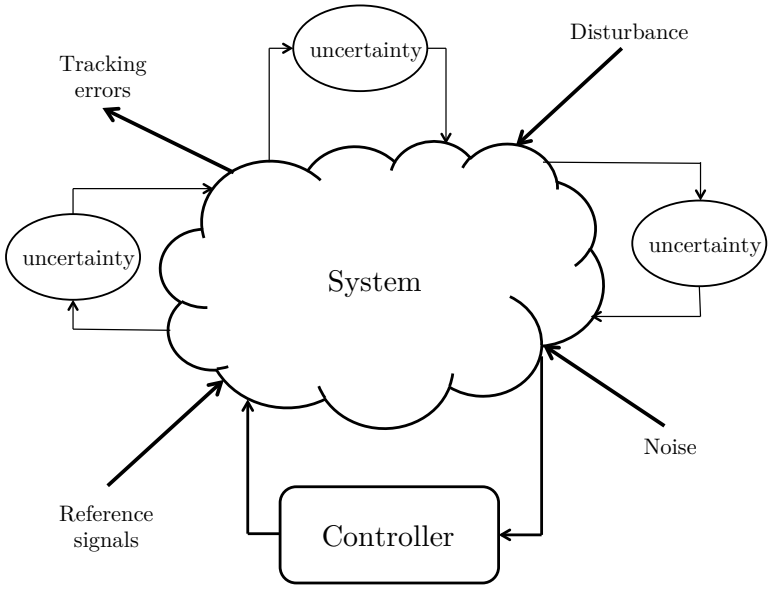


Figure 2.3: A practical control system representation

2.2 Representing Uncertainty

Uncertainty is at the heart of robustness concepts, and has been researched extensively [15][24]. As stated earlier, the environment of space inherently contains several uncertainties, e.g., in the form of disturbance forces and torques. In addition, the mathematical modelling process introduces uncertainty, for example, when linearisation is carried out, or when couplings or higher order effects are neglected. Further uncertainties arise due to physical effects too complex to model completely, such as effects of slosh. Additionally, there exist quantities, such as sensor errors, that are unknown or immeasurable. All of these must be carefully accounted for during analysis in order to gain insights into realistic performance of the control system. Such a control system representation, accounting for uncertainty, external disturbances and noise, is illustrated in Figure 2.3.

Uncertainty is sometimes classified as epistemic and aleatoric [25][26]. Epistemic un-

certainty is systematic, or knowledge uncertainty, and describes parameters that have precise values but which are unknown at the time of design. These may or may not be known with precision on launch day. Examples of such parameters are drag coefficient or engine efficiency. Aleatoric uncertainty is statistical or luck uncertainty, and describes parameters that have unavoidable stochastic variability, such as wind gust. It is important to account for both types of uncertainty in the design and analysis process. It is common to lump both types of uncertainty together; however, it may give rise to overly pessimistic or conservative results, and thus may not be an ideal strategy [26]. A good approach is, therefore, to attempt to reduce all epistemic uncertainties to aleatoric uncertainties via rigorous modelling efforts. A third type of uncertainty, error, is sometimes distinguished, which accounts for numerical effects such as rounding-off [27]. All of these may be informally called known unknowns. There are also "unknown unknowns", i.e., factors not accounted for because they are neither articulated not accessible [28].

Within the robust control community, uncertainty is classified into structured and unstructured uncertainty [15]. Unstructured uncertainty may include perturbations and uncertainties in different parts of the system. These can be lumped together and expressed as transfer functions or weights included in the block diagram during analysis. They often consist of effects in the high-frequency region. Unstructured uncertainty is also classified as additive and multiplicative [24]. Additive uncertainty includes additive plant errors, neglected high-frequency effects and uncertain right half plane zeros. Multiplicative uncertainty includes sensor and actuator errors. Since unstructured uncertainties may be included as weights, they can be operated on using classical control methods. Unstructured uncertainty was widely used to represent all model uncertainty before the development of robust control theory [29], but was found to be insufficient to account for all the effects necessary for modern control methods.

Thus, an important class of uncertainty, i.e., *parametric uncertainty* emerged. Plant perturbations may be considered as varying within a particular range of values (known or measurable), i.e., they may be represented as uncertain parameters [30]. Such uncertainty is consequently known as parametric uncertainty. When such uncertain parameters are collected from over the entire system and arranged together, they are observed to have structure [31]. When such uncertainties are analysed using traditional methods, they yield pessimistic results which exceed their true effects on the plant. Therefore, it is desirable to leverage the structure. This is done using the linear fractional transformation-based machinery and the so-called structured singular value [15], which are formalised in section 2.5. The studies in this thesis, unless stated otherwise, deal with parametric uncertainties, and the term uncertainty is understood to imply parametric uncertainty.

The modelling of uncertain parameters is now presented. Suppose an uncertainty (say, in the mass of spacecraft) is represented by δ_j for some index j, and is known to vary between $[\delta_{j_{min}}, \delta_{j_{max}}]$. Then, assuming the uncertainty is symmetric [32], its *nominal value* is $\delta_{j_0} = \frac{\delta_{j_{min}} + \delta_{j_{max}}}{2}$, and its *range* is $r_j = \frac{1}{2}(\delta_{j_{max}} - \delta_{j_{min}})$. The uncertainty is then represented as $\delta_j = \delta_{j_0} + r_j d_j$, where d_j represents a random parameter varying in [-1,1]. k uncertain parameters are represented together in the form of an uncertainty vector $\delta = [\delta_1, ..., \delta_j, ...\delta_k]^T$. Formally, then, uncertainties are represented as $\delta \in \mathbb{D} \subset \mathbb{R}^k$, where $\mathbb{D}$ is the uncertainty domain bounded[1] in [-1,1].

Before proceeding to a discussion of performance analysis techniques in the literature, the concept of performance is formalised in the following section.

[1] Uncertainties are normalised to lie in [-1,1] in many applications [33]. This thesis adheres to the same convention.

2.3 Performance Functions for Robust Control Analysis

In order for robustness analysis to be carried out, the stability and performance criteria must be suitably expressed. The performance function is defined for this purpose. For a closed loop uncertain system, the performance varies as the uncertain parameters vary. Recall that the uncertainties are $\delta \in \mathbb{D} \subset \mathbb{R}^k$. Then the performance is expressed as a function $J(\delta) : \mathbb{D} \to \mathbb{R}$. $J(\delta)$ is also known as the objective function, the cost function, or simply as the performance. The performance function is chosen by the engineer, and may encapsulate different measures of stability or performance. Performance in the classical control system context usually includes reference command tracking, disturbance rejection and noise rejection [15]. Various measures of robust performance exist, such as:

$\mathbf{H}_\infty$ norm: The H_∞ norm is a widely used measure, given by $J(\delta) = \|F_l\|_\infty$, where F_l is the lower linear fractional transformation defined in the following section. In single-input single-output (SISO) systems, the H_∞ norm coincides with the peak of the Bode plot magnitude of the transfer function. In multiple-input multiple-output (MIMO) systems, it is a more general measure that indicates the maximum energy gain of a system from U to Y [34].

$\mathbf{H_2}$ norm: The H_2 norm represents the average power of steady state output when the input is white noise [35]. It is therefore a good representation of the impact of disturbances on the system gain.

Sensitivity and complementary sensitivity functions: The sensitivity function represents the transfer function from the disturbance to output, while the complementary sensitivity function is the transfer function from the noise signal to the output

[15]. They are, respectively, indicators of disturbance rejection and reference tracking capabilities. Each of these may be measured by their gains in SISO systems, or singular values in MIMO systems, e.g., $J(\delta) = \bar{S}_v(S)$ where S represents the sensitivity function.

Time domain measures: These measures are specified in terms of the trajectory, and may depend on the problem under consideration [36]. Examples include terminal velocity, terminal position error and angle of attack.

Stability: Measures of stability are expressed in terms of the stability radius, or in terms of the gain and phase margin of the system [35]. Note that stability measures are included here for completeness, but are not utilised in this thesis.

2.4 Problem Formulation

Suppose a system's dynamical behaviour is represented by the following set of differential equations:

$$\dot{X(t)} = F\left(X(t), U(t)\right) \tag{2.1}$$

$$Y(t) = H\left(X(t), U(t)\right) \tag{2.2}$$

where $X \in \mathbb{R}^n$ are the states, $U \in \mathbb{R}^m$ are the inputs, and $Y \in \mathbb{R}^q$ are the outputs. The continuous time functions F and H may be linear or non-linear. The effects of uncertainty on the plant's behaviour is included in the form of uncertain parameters as follows:

$$\begin{aligned} \dot{X(t)} &= F\left(X(t), U(t), \delta(t)\right) \\ Y(t) &= H\left(X(t), U(t), \delta(t)\right), \quad \delta \in \mathbb{D}, \forall t \end{aligned} \tag{2.3}$$

The plant (2.3) is said to be *robust* if $\forall \delta \in \mathbb{D}$, the plant's stability and performance characteristics remain within specified acceptable limits.

Various approaches exist to determine whether plants are robust; some will be discussed in this chapter. Performance analysis approaches existing in the literature may be broadly classified as Analytical, Monte Carlo based, Optimisation-based and Probabilistic techniques.

2.5 Analytical Techniques

Analytical techniques are a powerful class of techniques that have a strong theoretical basis in linear control theory. The simplest analysis is typically one of assessing the gain and phase margins [37]. These classical metrics are implemented in industry to gain insight into the system in initial phases of design and analysis [38]. Such metrics are straightforward for SISO LTI systems, and are implemented channel-by-channel in MIMO systems.

Another approach is the grid point analysis method, in which the model (linear or nonlinear) is analysed at a number of grid points of the operating region. The stability and performance measures (such as gain and phase margins) are assessed at every grid point by locally linearising the model about an equilibrium point. This technique extends linear analysis methods to nonlinear models, and thus finds application in the aerospace industry at different stages in the design and analysis cycle. However, it suffers from some drawbacks. Firstly, the reliability of results from this method heavily depends on the engineer's familiarity with the model and the reasoning behind selecting the grid points. Secondly, there is no guarantee that the worst-case stability or performance value is obtained since the process occurs on a discrete domain. A finer grid increases

the confidence in the results. However, this would be at the cost of increased computational load, while there is still no true guarantee that the result is indeed the worst case. Further, additional errors are introduced by the process of linearisation.

The above methods, based on classical control and modern control theory, do not by design include the effect of uncertainty. Historically, uncertainty was included as small perturbations in the form of weighting matrices or by assuming a white noise input [23]. As a result, a systematic understanding of how uncertainty impacted a controller's performance was lacking. To overcome these challenges, robust control techniques were developed in the 1980s and 1990s, with uncertainty occupying a central role in synthesis and analysis [39]. These techniques, like the gain and phase margins, are predominantly frequency-domain based; however, unlike their classical control predecessors, they are well-suited for multivariable system analysis. Such techniques, however, impose requirements on how the model is formulated, however. The model is required to be in the form of a Linear Fractional Transformation (LFT). This modelling framework is first introduced, followed by a discussion on the analytical techniques.

2.5.1 LFT Modelling

LFT is a modelling framework associated with modern robust control theory [40]. Nearly all robust control synthesis methods work with LFT models. The primary role of LFT models is in modelling uncertainty in control systems. For a linear time invariant (LTI) system, the LFT model of a control system may be arrived at by separating into a purely uncertain block Δ and a purely certain block M — a process known as "pulling out" the uncertainties [24] — and arranging as in Figure 2.4. The LFT modelling framework has certain benefits: it facilitates manipulation using state-space like machinery, using constant matrix manipulation. In addition, the effect of uncertainty on the plant may be explicitly expressed and studied. A useful property is

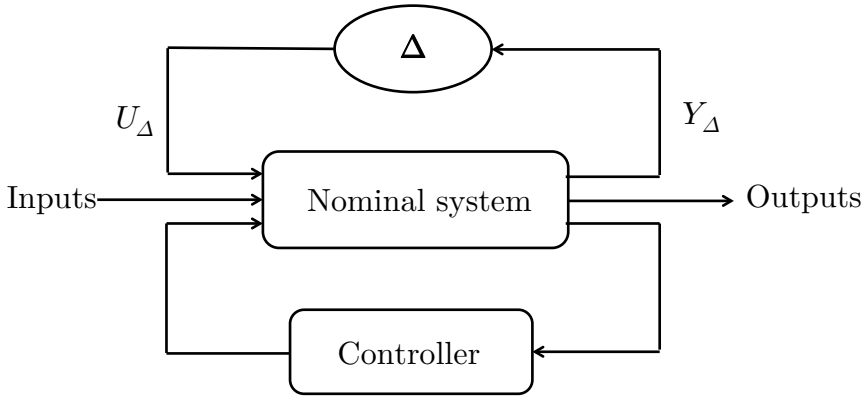


Figure 2.4: Control system representation with uncertainties lumped and pulled out

that any interconnection of LFTs is also an LFT [15].

Consider an LTI system expressed in the state space formulation, i.e.:

$$\begin{aligned} \dot{X} &= AX + BU \\ Y &= CX + DU \end{aligned} \tag{2.4}$$

where $X \in \mathbb{R}^n$ are the states, $Y \in \mathbb{R}^q$ are the outputs and $U \in \mathbb{R}^m$ are the controls. Adhering to standard control theory notation, A is the system matrix, B the input matrix, C the output matrix and D the feed-through matrix, with each matrix being of appropriate dimension.

In equations (2.4), Y represents the extraneous output (measurement) and U represents the extraneous input (such as reference signals and disturbance signals). In addition, there exist internal inputs W and internal outputs Z that attain significance in the presence of uncertainty. These appear in the LFT representation, after uncertainties are lumped together and pulled out using proper algebraic manipulations. They are related as $W = \Delta Z$. When Δ is diagonal, $W_1 = \delta_1 Z_1$, $W_2 = \delta_2 Z_2$, and so on.

Equations (2.4) may be rewritten including these internal inputs and outputs by partitioning the elements of the A, B, C, D matrices depending on how they are affected by the various inputs and outputs, as follows:

$$\begin{aligned} \dot{X} &= AX + B_1W + B_2U \\ Z &= C_1X + D_{11}W + D_{12}U \\ Y &= C_2X + D_{21}W + D_{22}U \end{aligned} \tag{2.5}$$

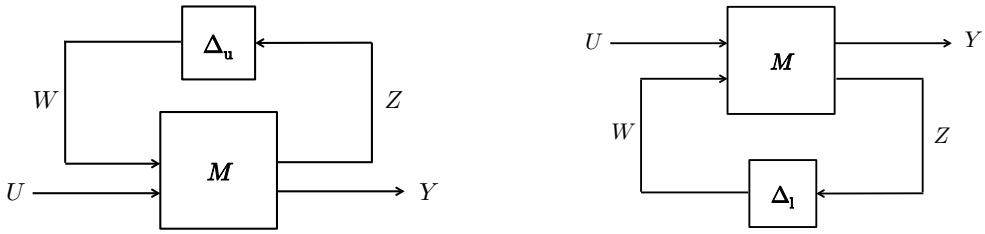
Equations (2.5) may be rewritten by separating the components of respective inputs and outputs in the following way:

$$\begin{bmatrix} Z \\ Y \end{bmatrix} = \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right] \begin{bmatrix} W \\ U \end{bmatrix} \tag{2.6}$$

where $M_{11} = \begin{bmatrix} A & B_1 \\ C_1 & D_{11} \end{bmatrix}$, and represents the effects of W on Z, $M_{12} = \begin{bmatrix} B_2 \\ D_{12} \end{bmatrix}$, and represents the effects of U on Z. $M_{21} = \begin{bmatrix} B_2 & D_{12} \end{bmatrix}$, which represents how W impacts Y, and $M_{22} = D_{22}$, which represents how U impacts Y, i.e., similar to the nominal case. M_{22} represents a nominal mapping which is perturbed by Δ, while M_{11}, M_{12} and M_{21} represent a prior knowledge of how Δ affects M, where $M = \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right]$[1].

M is known as the coefficient matrix, and, together with Δ, forms an LFT representation. Notice that M is entirely certain, while Δ is the uncertain component. Observe also that the dimensions of W and Z depend on the dimensions of M_{11} and M_{12}. LFTs

[1] Note that the partitioned matrix notation is left ambiguous in the literature [12], and may refer to a partitioned transfer matrix (as with M in this case), or to state space matrices as $\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$. Both meanings are used in this thesis, and it is assumed that the context makes the usage clear.



(a) (b)

Figure 2.5: Upper and lower LFT formulations

are classified depending on their structure as upper LFTs (F_u), such as in Figure 2.5a or lower LFTs (F_l), such as in Figure 2.5b. Their mathematical expressions, i.e., the transfer function from W to Z, are as follows:

$$F_u(M, \Delta_u) = M_{22} + M_{21}\Delta_u(I - M_{11}\Delta_u)^{-1}M_{12} \tag{2.7}$$

$$F_l(M, \Delta_l) = M_{11} + M_{12}\Delta_l(I - M_{22}\Delta_l)^{-1}M_{21} \tag{2.8}$$

provided that the inverse terms $(I - M_{11}\Delta_u)^{-1}$ and $(I - M_{22}\Delta_l)^{-1}$ exist. Such a formulation explicitly gives an idea of the effects of Δ_u (or Δ_l) on M. Figures 2.5a and 2.5b show the upper LFT and lower LFT interconnections respectively. Further, note that $F_u(N_u, \Delta_u) = F_l(M, \Delta_l)$ with $N_u = \begin{bmatrix} M_{22} & M_{21} \\ M_{12} & M_{11} \end{bmatrix}$.
If it is assumed that the uncertain parameters do not interact with each other, as is often the case [30], Δ_u (or Δ_l) can have a diagonal or block-diagonal structure. A block-diagonal structure occurs when one of the uncertainties, say δ_j, occurs more than once in the Δ block. If δ_j occurs twice, for example, the contribution of δ_j in Δ is $\delta_j I_2$, where I_2 is the identity matrix with 2 rows and 2 columns. The number of occurrences of an uncertainty depends on the complexity of the problem; it also depends on the modelling technique used. For large complicated systems, it is possible to have a non-minimal representation, i.e., one with more than the required number of occurrences of uncertainty. In such instances, minimal representations must be found, for which

techniques exist [41]. Non-minimal representations of LFTs add to the complexity of robustness analysis problems [42].

Building on the concepts of structured and unstructured uncertainty in section 2.2, if Δ_u (or Δ_l) has a diagonal or block-diagonal structure, it is called structured uncertainty. It takes this form in the case of purely parametric uncertainty. Structured uncertainty mainly includes physical parameters varying between bounds (usually known or easily estimated). The variation of these uncertainties follow a distribution, typically a uniform or normal distribution. Structured uncertainty may include interval, ellipsoid or diamond uncertainties [43]. In [29], the author points out that the structure of Δ_u depends on the performance objective and the uncertainty present in the system. Defining the structure consists of specifying the number of uncertainty blocks, the type of each block and the dimensions of each. If Δ_u is a full matrix, it constitutes unstructured uncertainty, or non-parametric uncertainty [11]. It can include:

- unmodelled dynamics,
- truncated high-frequency modes,
- nonlinearities such as delay, saturation and rate limits,
- initial values, and,
- effects of linearisation and time variation.

Having discussed the LFT modelling framework, some analytical techniques are now introduced.

2.5.2 Small Gain Theorem

The small gain theorem [24] states that for $\|\Delta_u\|_\infty < 1$, and given that M is stable, the $M - \Delta_u$ system is robustly stable if

$$\bar{S_v}(\Delta_u(s)) < \frac{1}{\bar{S_v}(M_{11}(s))} \tag{2.9}$$

where $\bar{S_v}$ represents the maximum singular value. For proof and technical details, an interested reader may refer [24]. The small gain theorem may be understood from the upper LFT representation in Equation (2.7). Since M is stable[1], it follows that the partitions M_{22}, M_{12} and M_{21} are stable. Since every term other than $(I - M_{11}\Delta)^{-1}$ is stable [44], the transfer function may be said to be stable if $(I - M_{11}\Delta_u)^{-1}$ is also stable. $(I - M_{11}\Delta_u)^{-1}$ becomes unstable when $M_{11}\Delta_u = I$. It follows that the magnitude of Δ_u must be smaller than the magnitude of M_{11}, which is what is expressed in Equation (2.9), since singular values are good measures of MIMO plant stability [31]. However, this criterion is conservative in the structured case (with many zeroes) [44]. Therefore, the concept of structured singular value (S.S.V.) is utilised. S.S.V was first introduced by Doyle in a seminal paper [31]. It provides a framework for both robust control analysis and synthesis.

2.5.3 μ-analysis

Structured uncertainty involves a diagonal or block-diagonal uncertainty matrix Δ_u. The subscript for upper LFT block are now dropped for readability. Consider a diagonal

[1] Robustness analysis is not carried out until nominal stability is first established

Δ matrix:

$$\Delta(s) = \begin{bmatrix} \delta_1 & & & & \\ & \ddots & & & \\ & & \delta_j & & \\ & & & \ddots & \\ & & & & \delta_k \end{bmatrix} \tag{2.10}$$

with $\Delta \in \mathbb{D} \subset \mathbb{R}^{k \times k}$ Then the structured singular value $\boldsymbol{\mu}$[1] is defined as follows:

$$\mu(\mathrm{M}) = \frac{1}{\min\{\bar{S_v}(\Delta) : det(I - M\Delta) = 0\}} \tag{2.11}$$

Now, scaling the Δ block to make $\bar{S_v}(\Delta) \leq 1$, we find that $\boldsymbol{\mu}(M)$ must be $< 1 \ \forall \ \omega$. This provides a criterion for stability and performance. The measure $\boldsymbol{\mu}(M)$ may intuitively be understood as the inverse of the smallest norm of the perturbation matrix Δ that destabilizes the $M - \Delta$ system. The structured singular value may be seen as a generalization of the maximum singular value. Indeed, it may be observed that $\boldsymbol{\mu}$ reduces to $\bar{S_v}$ in the presence of a single uncertainty. The sufficient condition for nominal stability (NS) is $\bar{S_v}(M_{11}) \leq 1 \forall \omega$. Nominal performance is guaranteed iff $\bar{S_v}(M_{22}) < 1$ $\forall \ \omega$. Robust stability is guaranteed iff $\boldsymbol{\mu}(M_{11}) < 1 \forall \ \omega$. Theoretical development of $\boldsymbol{\mu}$ is well-documented in the literature [31],[45],[46]. A detailed treatment of $\boldsymbol{\mu}$-analysis along with a historical perspective is presented in [29].

The robust performance (RP) problem is usually solved by reformulating as an RS problem. The solution is then found by introducing a fictitious uncertainty block Δ_p, representing the robust performance criterion, and which serves to interconnect M_{11} to the full plant. The fictitious uncertainty block Δ_p and the existing structured uncer-

[1]Note that the structured singular value is denoted as $\boldsymbol{\mu}$, in a different style than the standard notation, to distinguish it from the mean μ in probability based techniques discussed later.

tainty block may be combined using LFT interconnections to form a new uncertainty block Δ'. Then, the robust performance criterion is $\boldsymbol{\mu}_{\Delta'} < 1 \; \forall \; \omega$ [37].

Solving for $\boldsymbol{\mu}$ is a difficult problem; in fact, it belongs to a class of problems whose computational complexity is NP-hard (nondeterministic polynomial time) [47]. It has been found that it is intractable to calculate $\boldsymbol{\mu}$ exactly in large industrial systems with several uncertainties [48]. Thus, the solution is approximated by finding the upper and lower bounds for $\boldsymbol{\mu}$. In this case, the upper bound of $\boldsymbol{\mu}$ gives the sufficient condition for robust performance, while its lower bound provides the sufficient condition for instability.

The computational complexity of determining these bounds depends on the size of Δ, and whether Δ is real, complex or mixed[1]. Upper bound algorithms are finite frequency grid-based. Lower bound computation, in general, is harder. For purely complex or mixed real and complex Δ, polynomial time algorithms exist, such as power iteration [29]. For purely real systems, calculation of the lower bound of $\boldsymbol{\mu}$ is complicated. Several algorithms [45] have been proposed, including the gain method, which has better convergence properties than power iteration. However, the computation cost of the gain method grows exponentially with dimension of Δ, and so this method is limited to systems with low dimensional perturbation matrix.

One way to circumvent the issue is to add a small complex value (of the order 10^{-3} or smaller) to each uncertainty. This aids the algorithm in converging to a solution. A caveat is that the resulting solution is usually conservative. Matlab's Robust Control Toolbox has a good $\boldsymbol{\mu}$ implementation for both `robuststab` and `robustperf`. The computation is done by gridding in the frequency-domain, and determining whether

[1] Concepts of real and complex uncertainty are of relevance in the problem space, i.e., the when coefficients of the differential equation describing a system's dynamics are uncertain. Such coefficients may be real, complex or mixed [49]

pole migration occurs across the stability boundary as uncertainty varies[50]. The toolbox incorporates extensive research on robust control techniques from the literature.

2.5.4 Nonlinear analytical techniques

The above techniques are principally based on linear or linearised systems. Recent research on nonlinear system analysis has led to the development of techniques with some capabilities to analyse nonlinear systems. Certain classes of nonlinearities such as saturation can be incorporated into the LFT and analysed [36]. In such cases methods such as the Popov criterion [51] and multiplier methods may be employed. Other useful extensions which can handle certain types of nonlinear dynamics have also recently been developed; this include the Integral Quadratic Constraints (IQCs) [36] and Sum-Of-Squares programming [52]. In addition, Lyapunov stability theory based methods such as LMI are also in use [53].

The IQC method is a unifying framework for system analysis that generalises the small gain theorem and passivity theorem and other results from robust control. This analysis method allows a designer to handle different kinds of uncertainties, time invariant, time varying, delay and non-linear uncertainties simultaneously. The theory takes into account time varying uncertainties as well as their variation rates, leading to greater accuracy. The system to be studied is converted into a standard LFT form by pulling out the uncertainties, which then can be described by IQCs. The multiplier/IQC framework is based on a special modelling method where the analysed system is written as the LTI interconnection of sub-systems of different natures: i.e, sub-systems that are time-varying, non-linear, uncertain and those with time delays. Each subsystem is characterised by a set of multipliers or IQCs. In [36], the IQC-based analysis of NASA's HL20 re-entry is detailed. The study accounted for time-varying uncertainties in aerodynamic coefficients, apart from time-invarient uncertainties in the centre of gravity,

mass and inertia. It is reported that IQC analysis is an effective analysis method. An increase in computational complexity is seen for more complex problems. The authors point out the need for further research in LFT model reduction, efficient LMI solvers and computationally light IQC multipliers. Commercial software for IQC-based analysis is the IQC-β toolbox. The Integral Quadratic Separation (IQS) builds on IQC theory and converts the constraints into "separators", and is useful in descriptor systems.

Sum of Squares (SOS) techniques study the stability or performance of a dynamical system represented in a polynomial form. The method falls in a class of semi-definite programming problems which involves solving (linear convex) optimisation problems in the cone of positive semidefinite matrices. SOS programming can be used to guarantee stability and performance of control systems including non-linear systems, continuous/discrete hybrid systems, and time delay systems. Polynomial Lyapunov functions, including those of degrees higher than a quadratic Lyapunov function, are computed and tagged as certificates associated with the dynamical system. Hence, local stability and performance can be ensured in polynomial sub-level sets of the state space. The robustness analysis of an FA-18 model with a non-linear Dynamic Inversion based controller is conducted using SOS in [52]. SOS programmes are formulated for different characteristics, such as stability, input-output properties and system energy. The analysis effectively shows that as the uncertainty in the model increases, performance degrades.

Both robust stability and robust performance analyses can be performed via Linear Matrix Inequalities (LMI)-based methods. These methods naturally arise from the application of Lyapunov stability theory. Different approaches exist to define LMI-based stability and performance tests, which can be applied to uncertain systems:

- *The use of one Lyapunov function for all systems belonging to the uncertain set:* In most cases, the Lyapunov functions employed within this approach are restricted to those that can be expressed by quadratic forms of the state vectors of specific polynomial class as well polyhedral. Nevertheless, restricting the tests to quadratic stability implies imposing conservative conditions for robust stability/performance analysis.

- *Using a parameterized family of Lyapunov functions*: If parametric uncertainty is present in the system, these methods allow employing of parameter-dependent Lyapunov functions to assess robust stability and performance so that accurate analysis is possible. Nevertheless, the structure of the parameterization needs to be such that the resulting conditions are LMI representable.

A common feature of all of the above methods is that certain demands are made on the closed-loop simulation model under investigation, e.g., simplified versions of the full simulation model must be developed in order to generate LFT-based models for IQCs, and polynomial representations for the SOS methodology. These methods are still preliminary in terms of their application to industry models, although there is significant research interest in developing them further. However, the major drawbacks of these methods are the computational feasibility and the modelling effort required in formulating LFTs (for IQC, SOS and Popov based methods). The next section discusses Monte Carlo techniques, which are popular in the industry for $V\&V$ purposes primarily because they do not impose requirements on the model structure.

2.6 Monte Carlo Techniques

Monte Carlo methods have been used in various fields for more than a century. In the scientific literature, the seminal work is the 1949 paper by Metropolis and Ulam [54]. Monte Carlo-based statistical techniques are the most widely used techniques in

aerospace industries for verification and validation. They have several advantages: they can be applied to the full, high-fidelity, nonlinear model, and they can take into account uncertainties of all types i.e., initial conditions, time-varying, time delays, and so on. They do not impose any constraints on the structure of the model (or the uncertainty block). They require very little effort on the part of the engineer analysing the control design.

Various statistical approaches exist to determine the worst case performance and the associated combination of uncertain parameters. The gridding technique or vertex method [14] is one of the primitive techniques, and only evaluates the cost function at the extremes of the parameter space. This approach relies on the principle that as the uncertain parameter values increase in magnitude, performance necessarily degrades. Even presently, the state-of-practice for some some high-fidelity industry analysis is Vertex analysis [55][56]. Vertex approaches can work capably if the performance criterion and uncertain parameters are related linearly, which is impractical in large nonlinear systems with complicated performance function spaces. In vertex analysis, each uncertainty can take values of ± 1, and the cost is deterministically computed at every vertex case, and the worst case found from within the obtained performances. The computational complexity increases greatly as the number of parameters increase, i.e. as 2^k, and so the method becomes intractable for models where k is large. A more serious issue is that there is no guarantee that the extremes of the parameters cause the worst case.

It is often the case that poor performance occurs inside the parameter space, not at the extremes. Further, [11]. Thus, a widely implemented method is to randomly sample the parameter space, and to evaluate the cost function at each of these samples. The cost function values are then analysed, and the worst case, mean and standard devia-

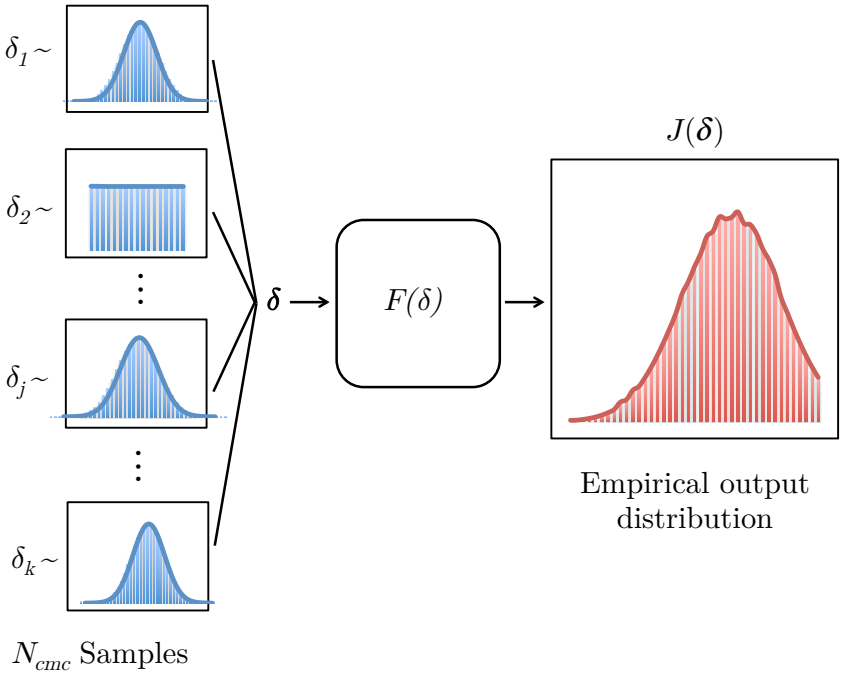


Figure 2.6: Schematic showing the Monte Carlo principle

tion of the costs computed. This is the basic idea of Monte Carlo-based performance analysis.

Figure 2.6 is a schematic of randomised Monte Carlo methods. Each uncertain parameter δ_j is assumed to be distributed according to a probability distribution. This could be, for example, a normal distribution or a uniform distribution. In the Monte Carlo method, random samples are drawn of each uncertain parameter from its distribution, which are then combined to form one uncertain parameter vector. Each combination of these samples is input into the plant, which is simulated with these uncertain vectors. A cost function is calculated at the end of the simulation and stored. This process may be carried out a number of times, say N_{cmc}. At the end of N_{cmc} evaluations, the cost function values are analysed. The worst-case cost may be found from these values. In addition, the empirical mean and standard deviation of these observed cost function

values may be found. This gives an empirical cost function distribution, which may be normal if the central limit theorem [57] is satisfied for the plant.

Monte Carlo is based on the weak and strong laws of large numbers. Formally, the principle is to generate N_{cmc} samples from the set $\mathbb{D}$, i.e. $\delta_1, ...\delta_i, ...\delta_{N_{cmc}}$ according to their density function, say $\delta \sim f_\delta(\delta)$, where each δ_i is called a multi-sample of cardinality N_{cmc}.

Next, $J_1 = J(\delta_1), ..., J_i = J(\delta_2), ...J_{N_{cmc}} = J(\delta_{N_{cmc}})$ are evaluated. The worst case is given by the maximum value of performance, i.e. J^*. It is possible to find the empirical mean performance $\mathbb{E}J_{mean} = \frac{\sum_{i=1}^{N_{cmc}} J_i}{N_{cmc}}$ and the empirical standard deviation. It is also possible to visualise the performance using various data visualisation tools. Monte Carlo methods are said to break the curse of dimensionality [58], since the number of samples necessary does not depend on the number of uncertain parameters.

Despite this advantage, a large number of samples — sometimes up to hundreds of thousands of samples [59] — may be necessary to reach a high degree of statistical confidence in the results. The computational effort increases rapidly as the search space, desired confidence and accuracy levels increase. This can severely limit the reliability of such an analysis. In addition, there is no guarantee that the worst case solution has indeed been found, as the solution may not lie in the set of samples used. For strong statistical guarantees that a true worst case has been found, a large number of simulations is required. This is often not feasible within the computational constraints currently faced by the industry. With high dimensional and complex nonlinear models, it becomes expensive to conduct evaluations for tens of thousands of combinations.

In [60], MC is applied to simulate a reusable launch vehicle in the terminal area energy-management mode. The run consists of 22 parameters and 1000 evaluations. The

analysis identifies that the mission success rate is 88 percent. In [59], Monte Carlo is applied to model spacecraft debris re-entry into Earth's atmosphere. The Monte-Carlo method is used in conjunction with the Taguchi method [61] to reduce the size of the parameters. The Taguchi method is a statistical strategy to determine parameters that have the biggest impact on the results, thereby reducing the computational load. The study resulted in the development of the Calima tool to simulate space debris re-entry. The tool accounts for 11 uniformly distributed uncertain parameters. The results are obtained with 200,000 Monte-Carlo evaluations. It should be noted that several industrial analysis tools often carry out Monte-Carlo analysis at this scale, which is the reason that V&V is considered to be an extremely expensive step.

[62] details the analysis of the X-43A flight, which was the first unmanned vehicle to fly at hypersonic speeds with a scramjet engine. Several performance metrics, such as acceleration, Mach number and dynamic pressure are analysed. In addition, scramjet engine thrust and moments are analysed. 286 uncertain parameters uniformly distributed within the modelled bounds are considered. Stress testing includes simulating beyond the modelled uncertainty bounds, i.e. it involves the so-called unknown unknowns [28]. The objective is to assess vehicle stability in the presence of modelled uncertainties. Several thousand MC runs are performed, with parameters varied independently of each other. Hardware in loop tests are carried out for validation. It is observed that MC works very well if the model is a good representation of the dynamics, and if uncertainty bounds are modelled correctly. It is pointed out that MC cannot handle unknown unknowns, i.e. unexpected events that are not modelled. The paper also discusses the programming challenges faced in carrying out an analysis of the scale of industry verification studies. Further examples of Monte Carlo-based performance analysis are [63], the 6DOF METEOR re-entry (consists of 57 uncertainties, and carries out 3500 simulations), and [64], which discusses the analysis of the Mars Lab entry,

descent and landing performance.

It must be emphasised that the studies discussed here have arrived at excellent results; however, the results have often come at the cost of high computational resource usage, and heavy engineer involvement. Several improvements exist over classical randomised Monte Carlo, such as Quasi Monte Carlo (Sobol, Halton, Faure and Lattices) [58] to deterministically generate well-distributed points.

A widely used Quasi Monte Carlo method is Latin Hypercube Sampling (LHS), which generates samples such that there is relatively good space filling. LHS ensures more even coverage of the sampling space by first evenly partitioning the cumulative distribution function curve, where the number of partitions equals the number of samples required, say N_{LHS}. Then, a sample is randomly selected from each partition of the cumulative density function curve. There are several instances of this implementation [65]. However, the quasi-random LHS samples need to be generated and stored prior to the run, which may not be efficient when accounting for data storage and retrieval considerations. The approach also suffers from the drawback of a lack of guarantee that the solution indeed lies in the sampled regions (there exists a trade-off between accuracy and grid size).

Recent research has also investigated the use of surrogate models in determining worst case performance and risk [18]. In this approach, approximation models are computed, which mimic the input-output behaviour of the simulation model. Modelling methods such as Kriging method and multivariate adaptive regression splines may be used to generate these surrogate models. Notably, the polynomial chaos method is used to generate first- and second-order surrogate models in [18]. Such models are shown to be significantly cheaper to evaluate, while also being fairly accurate. Monte Carlo simula-

tions of these models can then be feasibly carried out. However, the study acknowledges that higher order (such as third- and fourth- order) surrogate models, which provide higher accuracy, are computationally challenging to generate.

A powerful class of Monte Carlo algorithms are Markov Chain Monte Carlo (MCMC), which are widely used in Bayesian Inference of stochastic processes[66]. Such methods construct a Markov chain (leveraging the state transition matrix to model the dependence of a random event to the previous random event), such that its equilibrium probability distribution approximates the target distribution. Iterating over such a Markov chain at equilibrium, say N_{MCMC} times, can provide an accurate estimate of probability. Several MCMC algorithms exist, such as the Gibbs sampler, the Metropolis Hastings algorithm, and so on. However, arriving at a reliable approximation of a Markov chain equilibrium distribution may take several passes of data, particularly in high dimensional spaces. Further, being a Bayesian method, its performance relies on the prior distribution specified.

The focus of the MC-based studies discussed in this chapter has primarily been worst-case analysis, as well as empirical mean and standard deviation calculations. [26] gives an excellent treatment of Monte Carlo approaches used in spacecraft system requirements verification and the considerations involved. The authors introduce order statistics for interpreting the data, and the concepts of consumer risk (CR) and producer risk (PR).

In the following section, powerful numerical optimisation approaches for $V\&V$ are discussed, which overcome some of the drawbacks of MC.

2.7 Optimisation-based Techniques

Optimisation algorithms intelligently search the parameter space to find an optimum. As such, this makes them suitable for worst case analysis problems, with the verification problem reformulated as an equivalent distance maximisation or minimisation problem. Optimisation techniques have a long history in solving engineering problems, and have been applied to myriad domains [67]. In analysis of flight control systems, they have garnered interest in the last decade [14],[68],[69], [70], [71].

They have several advantages over traditional robustness analysis methods because they do not require the plant model to be formulated in a specific way. They can carry out analysis of systems that are linear or nonlinear, time independent or time-varying, and with uncertainties belonging to several classes such as time delays, initial condition uncertainties, and so on. Optimisation algorithms have significant improvements in computational load as compared to Monte Carlo methods [69]. In addition, they are straightforward to implement, thus reducing engineering effort. In fact, optimisation algorithms may be applied with little modification to the existing Monte Carlo analysis framework.

The general structure of optimisation for WCA is shown in Figure 2.7. In order to apply optimisation (or indeed, Monte Carlo) strategies, the performance criterion must be reformulated as an objective function.

Optimisation algorithms are often classified as local optimisation and global optimisation methods. Local optimisation algorithms are effective within some feasible neighbourhoods, particularly when there is a single optimum. They may consist of gradient-based methods and non-gradient-based methods, however they necessarily make use of

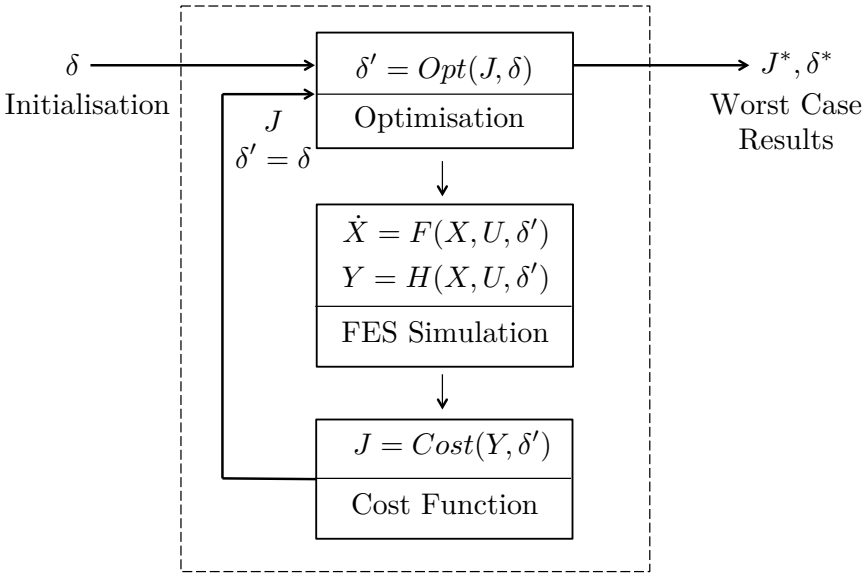


Figure 2.7: Schematic of optimisation-based worst-case analysis

knowledge about the parameter space, and solutions often depend on the initial conditions. Local methods converge to the optimum solution quickly. Sequential Quadratic Programming (SQP) is a highly effective local optimisation algorithm [11], which may be seen as a generalisation of Newtons method for unconstrained optimisation. In the SQP algorithm, the objective function is replaced by its quadratic approximation, and the constraints are replaced by linear approximations. SQP algorithms are capable of solving smooth and well-scaled non-linear optimisation problems when the functions and gradients can be evaluated with high precision. If the local gradients are not available analytically, numerical approximations are computed; this might reduce speed and accuracy, especially if the function evaluations are noisy. Matlab's useful `fmincon` optimiser is an implementation of SQP. A drawback of such local methods is that they may get stuck in a local optimum, particularly when they utilise a gradient to optimise. Local optimisation algorithms are well suited to convex problems.

Global optimisation methods overcome the local optimum issue. They can find a global

optimum over all feasible regions even in non-convex problems. They may be either deterministic or randomised. Deterministic optimisation methods systematically search the parameter space and provide guarantees about the solution arrived at. Examples are nonlinear programming and dividing rectangles (DIRECT). Randomised optimisation algorithms, on the other hand, rely on random samples of the parameters, and provide the optima in probabilistic sense. Examples of these are: genetic algorithms, differential evolution and simulated annealing.

Genetic Algorithms (GA) are stochastic search and optimisation algorithms, based on the evolutionary principle of "survival of the fittest" in a population that has evolved over a certain number of generations. A fitness function is defined to assign a performance index to each candidate. In genetic search techniques, a randomly sourced population of candidates undergoes a repetitive evolutionary process of reproduction through selection for mating according to a fitness function, and recombination via crossover with mutation. A complete repetitive sequence of these genetic operations is called a generation. Genetic algorithms have been applied in flight control clearance successfully in [69][72][18]

Although global optimisation methods are more capable than local methods, they often come at the cost of increased computational load. Particularly in the case of deterministic global methods, the computational resources required may be enormous. Stochastic methods generally fare better, with far fewer function evaluations required for the solution to be found. However, there are no formal convergence proofs[1], although they have been demonstrated to perform well in a wide variety of problems. The computational resources required, however, are higher for stochastic global methods than for local methods.

[1] An exception is the simulated annealing technique, which is proved to converge to the optimum at the limit [73].

Hybrid optimisation techniques take advantage of the benefits of both global and local optimisation methods. They incorporate a local search when the global algorithm fails to bring an improvement over a certain (pre-selected) number of iterations. Some examples are hybrid GA, hybrid DE and hybrid DIRECT [18].

Optimisation algorithms are shown to be useful in aerospace control performance analysis, and a combination of techniques sometimes yields beneficial results [68]. Since optimisation methods vary in their structure and implementation, they require considerable engineering effort to implement in industry clearance problems. The WCAT-II is a toolbox developed at the University of Exeter [18], consisting of a pool of optimisation algorithms in a consistent framework well-suited for aerospace control system analysis. It consists of local algorithms such as SQP, global algorithms such as DE, GA and DIRECT, and their hybrid counterparts. It has been implemented in several clearance problems [68],[74],[75]. The structure of WCAT-II is similar to the optimisation framework shown in Figure 2.7. Its benefits lie in addressing several performance problems unique to aerospace flight clearance, while being versatile to deal with various types of industry models.

In this work, the WCAT-II implementation of Differential Evolution is used in the analysis. DE is discussed in the following section.

2.7.1 Differential Evolution

Differential Evolution (DE) is a global optimisation technique [76] belonging to the class of evolutionary algorithms. It has been found to be very effective in arriving at the global optimum in a wide range of fields — [77] provides an overview of its appli-

cations. It has been used in flight control clearance and found to have a high efficacy. It has been applied to high-fidelity real life missions and has been effective in computing instances of failure as well as worst-case scenarios. The DE search consists of the following four steps: initialization, mutation, crossover and selection.

The scheme of DE used in this work is *DE\rand\1\bin*, which represents that the uncertainty vector to be mutated is randomly chosen, and that a single difference vector is used. A brief description of DE follows, and an interested reader is referred to [76] for a detailed treatment.

Initialisation: This step generates an initial population with N_p members. Each member is a vector $\delta = [\delta_1, \delta_2, ..., \delta_j ..., \delta_k]^T$. Each δ_i is generated in the following way:

$$\delta_i = \delta^L + \delta^U + \psi_i \left(\delta^U - \delta^L\right), i = 1, 2, ...N_p \tag{2.12}$$

where δ^U is the upper bound of the parameter, and δ^L is the lower bound. ψ_i is a randomly generated vector with elements in [0,1]. It is noted that this scheme corresponds to sampling randomly from a uniform distribution. It is straightforward to sample instead from any other distribution.

Mutation: In this step, a weighted difference between two randomly selected candidates δ_{r1} and δ_{r2} is added to another randomly selected candidate δ_{r3} to generate the mutated vector. The difference is $D_{r12} = \delta_{r1} - \delta_{r2}$. If C_m is the mutation coefficient, then the mutated vector is given by

$$\bar{\delta}'_m = \delta_{r3} + C_m D_{r12} \tag{2.13}$$

Observe that the direction of the difference vector D_{r12} indicates the current search di-

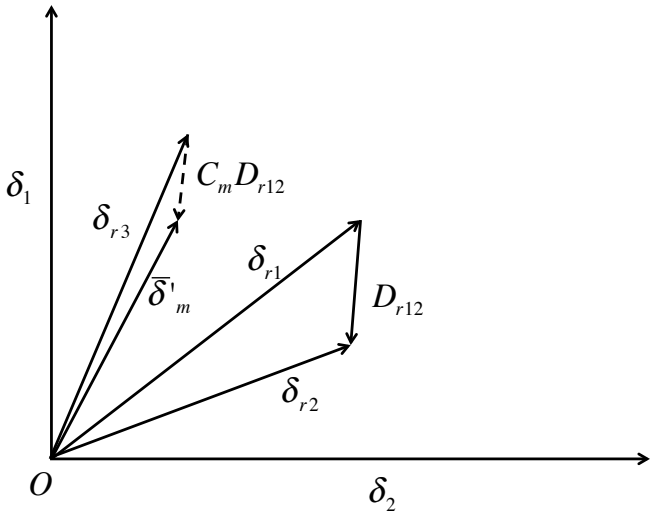


Figure 2.8: Schematic showing the Mutation operation in DE

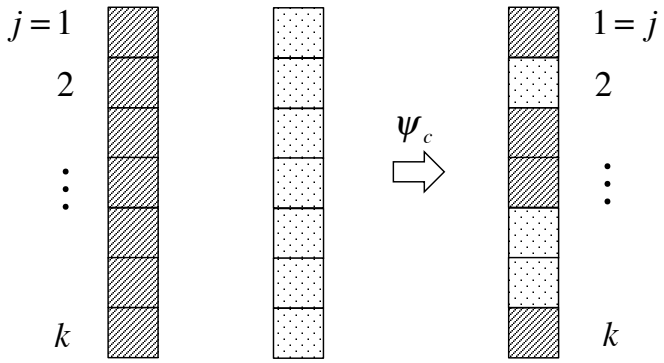


Figure 2.9: Schematic showing the Crossover operation in DE

rection, while C_m represents the step size. The mutation coefficient is chosen between 0 and 1.

Crossover: This step increases the diversity of the population. The mutated vector $\bar{\delta}'_m$ and a chosen parent individual δ_m are mixed element-by-element to form the trial vector δ'_m, depending on a randomly generated crossover constant ψ_c.

Selection: The trial vector δ'_m and parent vector δ_m are both used to evaluate the

value of the cost function. The vector that gives a lower cost is selected to be part of the new population.

Termination: The algorithm terminates after repeating the above four steps until the terminating criterion. There are several strategies for selecting the terminating criterion. An adaptive scheme is one where the algorithm terminates if there is no improvement over a specified number of iterations. In a fixed scheme, the algorithm terminates after a maximum number of allowed iterations.

A drawback of optimisation techniques is a lack of information about the probability associated with the worst case. Often, the worst-case performance computed may be conservative (due to conservative modelling of uncertainties), and not realistic practically. Having a knowledge of the probability or risk of the occurrence of worst case, can provide insight into the performance and guide the designer as to whether tuning or redesign of the control is necessary. The probabilistic techniques discussed in the following section aim to compute the risk associated with degraded performance.

2.8 Probabilistic Techniques

In order to complement the above techniques, probabilistic techniques have been researched in recent years. These techniques are centred around the notions of probabilistic robustness [78]. The primary drivers are to quantify conservatism and to reduce computational load. In addition, these techniques also benefit from being randomised, and therefore do not impose constraints on the structure of the problem, and may be readily implemented in existing industry frameworks (of Monte Carlo). These approaches seek to combine worst-case bounds with probabilistic information, so as to overcome issues of conservatism associated with the H_∞ design paradigm. In this way,

both probabilistic and deterministic aspects of a system's performance may be accessed. Researchers argue that probabilistic methods of synthesis and analysis may lead to control designs that are safer than those designed using purely deterministic concepts [79].

The first ideas of probabilistic methods in control systems appear in a 1980 paper by Stengel [80], which dealt with concepts of probability with regard to system stability. Renewed interest in this direction has gathered in the last two decades [81] [82] [83][84]. There has been some research interest in the subject since then. Some probabilistic analysis algorithms are: probabilistic $\boldsymbol{\mu}$-analysis, Monte Carlo-based analysis and safe and unsafe regions[18]. Several algorithms have also been applied for probabilistic control design synthesis, but these are beyond the scope of this thesis.

In an early paper [81], the treatment primarily involves discussing the applicability of statistical methods for robustness analysis, and discussion of the number of samples necessary to generate accurate results. Two bounds are primarily identified for the purpose. If ϵ and c are positive scalars representing the accuracy and confidence in the solution respectively, the Bernoulli bound [81] gives the number of samples required to achieve the solution to be:

$$N_B \geq \frac{1}{4\epsilon^2 c}, \tag{2.14}$$

while the Chernoff bound [32] provides the following expression:

$$N_{Ch} \geq \frac{log(\frac{2}{c})}{4\epsilon^2} \tag{2.15}$$

In addition, a third bound called the worst-case bound proposed in [82] is useful. Compared to the Bernoulli and Chernoff bounds, which deal with expectations, the

worst-case bound is particularly useful in the estimation of extrema. The worst case bound is given by:

$$N_{wc} \geq \frac{log\frac{1}{c}}{log\frac{1}{1-\epsilon}} \tag{2.16}$$

It is pointed out in [81] that the Chernoff bound outperforms the Bernoulli bound in terms of the number of samples needed. The worst case bound outperforms both of these by nearly three orders of magnitude. This indicates that the computational load issue may not be as dire as previously thought, albeit still worthy of consideration. A comparison of the three bounds is made with $c = \epsilon$ and is shown in Figure 2.10. In addition, it is noted that the number of samples in all three cases is independent of the type of distribution, the number of uncertain parameters, and the bounds on the uncertain parameters. It only depends on the accuracy and confidence levels required.

Some researchers argue that for applying probabilistic methods to rare events, which the worst case is expected to be, confidence and accuracy are not appropriate, and a measure known as the relative error should be used instead [66]. This measure will be utilised in later chapters of this thesis.

Several research studies have since been carried out on probabilistic robustness analysis with applications to aerospace control systems. It is generally understood that while there is tremendous benefit to their implementation, the development is still preliminary with much room for improvement. The book [58] gives an authoritative discussion on probabilistic robustness analysis and related research directions. It considers probabilistic robust analysis as well as design techniques. Some concepts introduced the probabilistic counterparts of classical robustness measures, such as probabilistic robust stability radius, worst-case, performance radius, probability degradation function, and

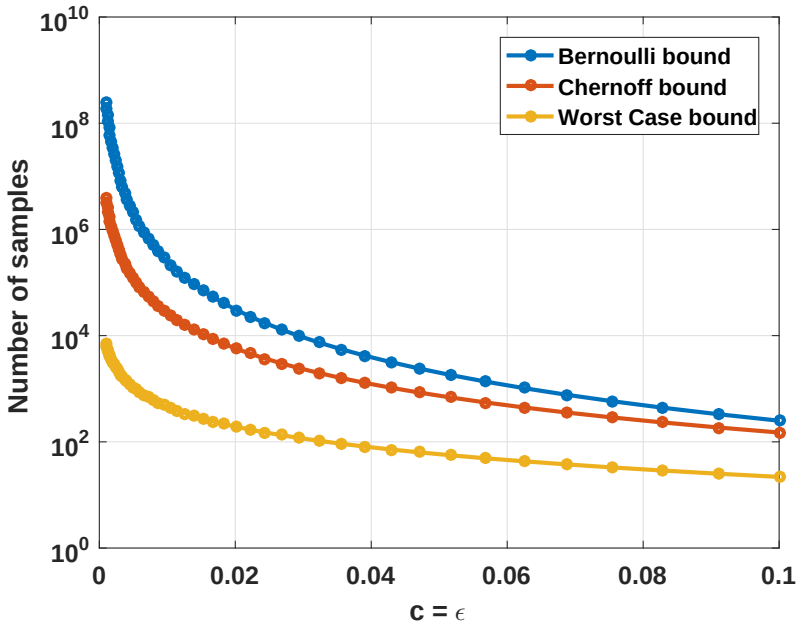


Figure 2.10: Comparison of number of samples required according to the Bernoulli, Chernoff and worst case bounds

safe regions.

The results are developed on the basis of the so-called "good set" $\mathbb{B}_G$ and "bad set" $\mathbb{B}_B$, which are subsets of the uncertainty domain $\mathbb{B}_D$. If the uncertainty set is $\delta \in \mathbb{D}$, the performance function is $J(\delta)$ and a desired level of performance is γ, the probability of achieving that level of performance is given by

$$p(\gamma) = P[J(\delta) \leq \gamma] \tag{2.17}$$

The good set $\mathbb{B}_G$ is the set of uncertain parameters that satisfy the performance criterion $J(\delta) \leq \gamma$, and is formally defined as

$$\mathbb{B}_G = \{\delta | \delta \in \mathbb{B}_D : J(\delta) \leq \gamma\} \tag{2.18}$$

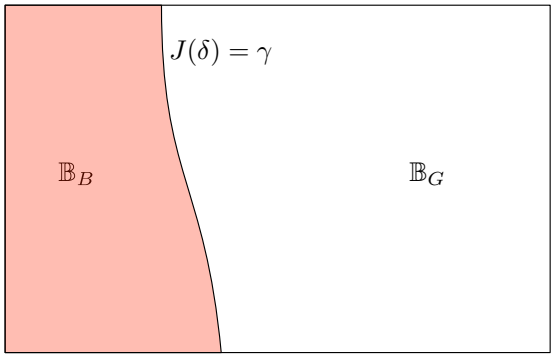


Figure 2.11: Good and bad sets

The bad set is complementary to the good set, i.e. it is the set of uncertain parameters that do not satisfy the performance criterion, i.e.,

$$\mathbb{B}_D = \{\delta | \delta \in \mathbb{B}_D : J(\delta) > \gamma\}, \tag{2.19}$$

An example of the good and bad sets is shown in Figure 2.11. For robust performance (or stability) to be guaranteed, $\mathbb{B}_G = \mathbb{B}_D$, and $\mathbb{B}_B = \Phi$, where Φ represents the null set. In the probabilistic sense, the criterion $J(\delta) \leq \gamma$ is to be fulfilled with some probability (or risk) $p < p_{critical}$. The probability of satisfaction is given in terms of the volumes of the good and bad sets:

$$p = \frac{Vol(\mathbb{B}_G)}{Vol(\mathbb{B}_D)} \tag{2.20}$$

In this context, the probabilistic worst case problem is written as follows: Determine a performance $\bar{\gamma}$ smaller than the worst case performance γ^* with a small probability, i.e., determine:

$$\bar{\gamma} \leq \gamma^* \tag{2.21}$$

$$P[J(\delta) \leq \bar{\gamma}] < p_{critical} \tag{2.22}$$

where $p_{critical}$ is the acceptable risk threshold of the performance exceeding $\bar{\gamma}$. In this thesis, if a system satisfies the performance criterion in Equation (2.23), *probabilistic robust performance* is said to be satisfied. Note that in this thesis, the convention is that higher performance corresponds to poor performance, and so the performance inequality is reversed. Additionally, the problem is stated in terms of the desired performance γ_{des}. Therefore, in this work, the probabilistic analysis problem is to guarantee that:

$$P[J(\delta) \geq \gamma_{des}] < p_{critical}. \tag{2.23}$$

It is intuitive that satisfying deterministic robust performance guarantees that probabilistic robust performance is also satisfied. However the inverse is not necessarily true. Similar interpretations exist for other probabilistic robustness measures.

In [58], the authors caution against arbitrary selection of uncertain parameter probability distributions. They demonstrate the issue using a truncated Gaussian distribution with mean centred at 0, with two different standard deviations, $\sigma = 0$ and $\sigma = \infty$. In the former case the distribution tends to an impulse function, while in the latter it tends to a uniform distribution. Intermediate distributions may vary anywhere between these extreme values, and therefore may not be meaningful. An interested reader is referred to [58].

A recent paper [85] discusses the development of probabilistic gain or probabilistic $\boldsymbol{\mu}$ [84] as a method to carry out probabilistic robustness analysis. In the work, analytical expressions for computing the upper and lower bounds on the cumulative distribution function of the performance function are derived. The authors argue that although probabilistic robustness analysis may incur extra computational cost, it provides valu-

able information regarding the conservatism of worst case bounds, and thus must be considered. The method developed is applied to NASA's generic transport model with both uniform and Gaussian distributions. The results provide convincing insights into system performance. The analysis is carried out for both SISO and multivariable cases. The studies on probabilistic $\boldsymbol{\mu}$ have typically implemented Monte Carlo analysis [86], while the need for an efficient algorithm to simulate the system, particularly in the tails of the distributions, is acknowledged [85]. The studies also comment on the benefit of probabilistic techniques to complement traditional worst case concepts to lower the likelihood of missed detections.

Another study [87] considers the probabilistic robustness analysis of an F-16 fighter plane, with either an LQR and a gsLQR controller. The authors discuss the concept of "distributional robustness" as opposed to the classical notion of robustness [88],[32]. They discuss the need for probabilistic techniques as a way to deal with the non-linear models and controllers of the F-16, as well as for time varying uncertainties. The authors point out that probabilistic models are useful for carrying out analysis without imposing structure in a model. Another work by the same team [89] incorporates probabilistic methods in a robustness analysis of jump linear systems. They utilise the Wasserstein distance as a means to compare the outputs of two models. The technique is demonstrated on an inverted pendulum stochastic jump linear system. The Wasserstein distance is not straightforward to compute, however, and is yet to be tested on high-fidelity systems. In addition, nonlinear models cannot be readily analysed using the proposed method.

The studies discussed show the value of probabilistic robustness analysis in $V\&V$. Such methods have the ability to quantify the probability of occurrence of worst case, identify problematic parameter regions, and relax conservative deterministic bounds. However,

the research area has not attained maturity. The applications covered in this thesis are high-fidelity industry systems as part of ongoing projects collaborating with industry engineers. For this reason, it is important to provide "cookbook solutions" that do not require further engineering effort before being implemented each time. The developed techniques must fit into the existing frameworks (e.g. the Monte Carlo framework currently used widely). It must, however, still be flexible enough to be applied to a variety of plants and controllers, and take into account different types of uncertainties.

2.9 Conclusion

As a vital part of $V\&V$ activities, robust performance analysis methods play a crucial role in the synthesis of spacecraft controllers. This chapter aims to form a foundation upon which the rest of the thesis builds. It discusses some approaches to robust performance analysis in the literature, from the classical analytical techniques to more recent approaches such as statistical and optimisation-based approaches. It also explores the recently developed probabilistic robustness analysis methods. This brief survey of the literature clearly demonstrates a need for performance analysis approaches that are accurate, efficient and implementable within existing $V\&V$ paradigms. Specifically, the importance of probabilistic measures of worst case performance is discussed. This thesis offers a methodology for carrying out probabilistic robustness analysis, i.e. using cross-entropy methods, which is detailed in the next chapter.

Chapter 3

A Cross Entropy-based Simulation Technique for Probabilistic Analysis

3.1 Introduction

Previous chapters have established the importance of assigning probability information to robust performance analysis results. This chapter develops a methodology for estimating such probabilities. In this chapter, the cross entropy technique for efficient estimation of the probability of rare events is described. An algorithm is developed to characterise the degradation of the probability of a performance metric of robust controller exceeding its desired performance threshold level, i.e., to characterise the probability profile of the system.

The cross entropy based simulation method was first introduced in [90] to compute the probabilities of rare events in complicated stochastic networks. The algorithm has since evolved to become the basis of a powerful set of techniques such as combinatorial

optimisation [91], continuous multi-extremal optimisation [92] and of course, rare event simulation [93]. It has since been applied to several problems in varied fields such as network reliability [94], path planning [95], and power systems [96].

The CE algorithm itself is a variance minimising importance sampling method. It adaptively reduces the "distance" between the parameter distribution and an ideal distribution. The distance measure is the Kullback-Leibler divergence [97], and is also called the cross entropy between two distributions.

This chapter describes the technique in some detail. First, some preliminary concepts of probability are reviewed. Then, the problem formulation is presented. The probability profile is presented and some of its properties are discussed. The cross entropy methodology is presented, followed by the algorithm. The algorithm used to generate the probability profile is then presented.

3.2 Preliminary Concepts

Some fundamental concepts of probability theory are reviewed in this section. Probability is, in general, expressed in the context of a **random experiment**, which is an experiment whose outcomes can differ even though it is performed identically every time [98]. An **event** constitutes a particular subset of the set of all possible outcomes of a random experiment. To an event, a probability of occurrence may be associated. A **rare event** is an event which has a low[1] probability of occurrence, (10^{-6} or lower). A **random variable** is a measurement (or observation) of a random experiment [13]. In [100] a random variable is defined somewhat intuitively as a variable that "takes

[1]The exact probability for an event to be defined as rare is not consistent in the literature, with some researchers [66] stating it is less than 10^{-9}, and some others stating a threshold of 10^{-3} [99].

on values by chance". Random variables are usually denoted by capital letters in the literature, say X, while their measurements are represented by the corresponding lower case letter x. However, since this thesis seeks to utilise probability theory concepts within the robust control analysis paradigm, this standard notation is modified while respecting the underlying meanings. The notation is introduced in the following section.

3.2.1 Incorporating Random Variables in the Control Analysis Framework

This work treats uncertain parameters of a model as random variables. These random variables are denoted by δ, while their observations are denoted by d. δ represents the uncertain parameter vector consisting of the k uncertain parameters $[\delta_1, \ldots \delta_j, \ldots, \delta_k]$. The remainder of the preliminary concepts shall be introduced adhering to this modified notation.

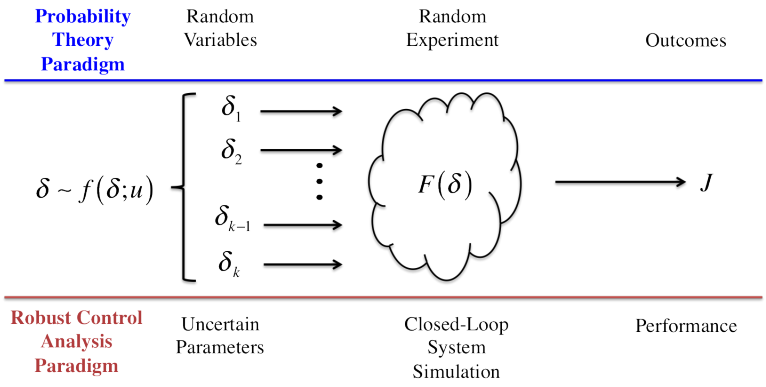


Figure 3.1: Drawing parallels between the paradigms of probability theory and robust control analysis

The uncertain parameters are considered to be **continuous** random variables, meaning that they can take "uncountably infinite" values [98]. Recall that any function of a random variable is also a random variable. Therefore the performance function is

also a random variable. The **Probability Density Function** (pdf) quantifies how a continuous random variable is distributed, i.e., what the likelihood of a random variable taking a particular value is. The pdf is denoted by the function $f(\delta; u)$, where u is the parameter of the distribution. The notation $\delta \sim f(\delta; u)$ is used to represent that δ is distributed according to $f(\delta; u)$. u may be a scalar or a vector, depending on the distribution. For instance, u may represent the rate parameter in the exponential distribution pdf $f(\delta; u) = ue^{-u\delta} \ \forall \ \delta \geq 0$. An example of u being a vector is in the uniform distribution pdf $f(\delta; u) = \frac{1}{b-a} \ \forall \ \delta \geq 0$ where a and b are the lower and upper bounds for the random variable δ.

Briefly reviewing the standard uncertainty representation in the robust control analysis framework, it is recalled that the uncertain parameters are normalised to lie in $[-1, 1]$ [15]. For example, say m is one of the uncertain parameters, representing the mass of a satellite. The mass may be known to vary between 2000kg and 2400kg during flight — due to fuel mass variation and other considerations known to system experts. Then, the **nominal value** of the parameter may be written as $m_0 = \frac{2000+2400}{2} = 2200$. The uncertain parameter can vary by ± 200 from m_0. This is formulated as $m = m_0 + r_m \delta_m$, where r_m is the **range** of the uncertain parameter (200 in this case), and δ_m is the normalised uncertain component which lies in $[-1, 1]$.

The random variables are considered to be: (i) independent of each other and (ii) identically distributed — this is commonly abbreviated as i.i.d. [13]. In other words, all the δ_js are distributed according to the same pdf f with the same mean μ and standard deviation σ.

In this chapter, the random variables are distributed according to a truncated normal distribution. The nature of the uncertain parameter distribution is defined by the

consortium of industry partners. This choice is attractive, because it incorporates beneficial properties of both the normal and uniform distributions, and is briefly discussed in the following section.

3.2.2 The Truncated Normal Distribution

The normal (or Gaussian) distribution is justifiably the most prevalent distribution in probability theory and statistics, since a large number of natural phenomena are observed to be normally distributed [98]. The Central Limit Theorem indicates that distribution of any average tends to be normally distributed [57]. Normally distributed random variables have higher probabilities of taking values close to the mean value, and lower probabilities away from the mean, as illustrated by the bell shaped curve. The normal distribution is also convenient in the current analysis since it leads to an elegant analytical result, discussed later in this chapter. The normal distribution is represented by the pdf in Equation (3.1)

$$f\left(\delta;[\mu,\sigma]\right)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(\delta-\mu)^2}{2\sigma^2}} \tag{3.1}$$

However, unless truncation bounds are specified, the resulting worst cases can be extremely pessimistic (since in the normal distribution the tails extend to $\pm\infty$).This avoids the pitfalls of the Law of Truly Large Numbers — that given a very large sample, any outrageous event is likely to occur [101]. Moreover, in reality the uncertain parameters are often limited. Further, the uncertain parameter normalisation already discussed requires the variability to be between -1 and 1 for each parameter.

Taking these factors into account, the distribution selected has a mean 0, and a standard

Table 3.1: Error due to truncation of the normal distribution, f_t, relative to the normal distribution f_t

δ	$f_o(\delta; [\bar{\mu}, \bar{\sigma}])$	$f_t(\delta; [\mu, \sigma])$	Truncation error = $\frac{f_o - f_t}{f_o}$
0.2151	0.8631	0.8740	0.0126
0.7336	0.1856	0.1879	0.0126
-0.5231	0.4241	0.4295	0.0126
-0.9035	0.0778	0.0788	0.0126

deviation 0.4, and is truncated at -1 and +1. The industrial rationale for this choice is the following:

- Despite the truncation, there exists a large probability of samples being drawn from within the [-1,1] bounds. The bounds corresponds to 2.5 standard deviations away from the mean on both sides, meaning that 100 - 2*0.6 = 98.78% of the samples drawn from the normal distribution lie within the bounds.
- Further, the choice still ensures that the probabilities at the extreme values [-1,1] are not unrealistically low — substituting $\mu = 0$, $\sigma = 0.4$ and $\delta = \pm 1$ in Equation (3.1) results in a probability of 0.044.
- Moreover, the error in the pdf value due to truncation is small. This is evidenced in Table 3.1, where calculations of the truncated pdf are done for a few values of δ. The relative error between the truncated and original densities is calculated, which is found to be the same value 1.26% for all cases.

These error values are considered by the consortium to be acceptable for the purposes of the analysis. Additionally, although the multivariate case is dealt with, the distributions reduce to multiple univariate densities since the parameters are independent, meaning that the interaction between parameters is not accounted for (i.e., the covariance matrix is diagonal).

The formulation of the robust performance analysis problem is done in the next section.

3.3 Problem Formulation

The optimisation-based worst case analysis problem is first reviewed. For a closed loop dynamical system, let

$$J(\delta) : \mathbb{D} \to \mathbb{R}$$

be a performance criterion, where $\mathbb{D} \subset \mathbb{R}^k$ is the bounded uncertainty domain. A desired threshold for each criterion may be written as $J(\delta) \leq \gamma_{des}$, where J is the performance criterion (also known as the cost function, or simply the cost) under consideration.

Following the approaches in [74], [102]and [103], the worst case analysis problem can be formulated as a maximisation problem (typically maximising a distance or norm value) subject to the dynamics of the model:

$$\begin{aligned} \max_{\delta \in \mathbb{D}} \quad & J(\delta) \\ \text{subject to} \quad & \dot{X} = F(X, U, \delta), \ \delta \in \mathbb{D}, \end{aligned} \tag{3.2}$$

where $X \in \mathbb{R}^n$ are the states, $U \in \mathbb{R}^m$ are the control inputs, and F represents the dynamics of the system. Suppose the worst case performance obtained is γ^*.

The objective is to determine the probability that the performance $J(\delta)$ exceeds a given value (or "performance level") γ, as γ degrades. i.e., the probability of the event $J(\delta \geq \gamma)$ is to be calculated for various values of γ, particularly:

- $\gamma = \gamma_{nom}$, the nominal performance value, with all parameters set to 0. This probability is expected to be very high (≈ 1).

- $\gamma = \gamma_{des}$, the performance criterion. It is desired that this probability is low, to indicate that there is no violation of this criterion for a majority of cases. However, in practice this may be quite high.

- $\gamma = \gamma*$, the worst case performance. This is considered to be a rare-event, and the probability is expected to be very low, of the order 10^{-6} or lower [13].

Taking into account the expected trends, a plot of the probability associated with the event $[J(\delta) \geq \gamma]$ vs. γ may be constructed — this is shown in Figure 3.2. This curve is called the *Probability Profile of Performance*(PPoP), and constitutes one of the central ideas of the thesis. This curve is similar to several analogous ones such as the survival curve showing effects of disease in medical science, the titration curve in chemistry, the phase transition curve of populations in ecology and the survival curve in reliability science showing machine wear and tear. In the context of robustness analysis, the PPoP represents the transition from nominal performance to problematic performance of spacecraft controllers as uncertain parameter values vary. The notion of probabilistic robustness may be interpreted from Figure 3.2. If the probability that performance is greater than γ_{des} is below a critical level, probabilistic robust performance is said to be achieved. The next section examines the PPoP curve further and discusses some of the insights it offers.

3.4 Properties of the Probability Profile of Performance Curve

From Figure 3.2, two points of particular interest are identified. Most importantly, the probability associated with the worst case performance level $\gamma*$ is a small value $p*$. This is the crucial probability associated with the worst case. In addition, the probability associated with the desired performance level γ_{des} is found to be p_f, sometimes called

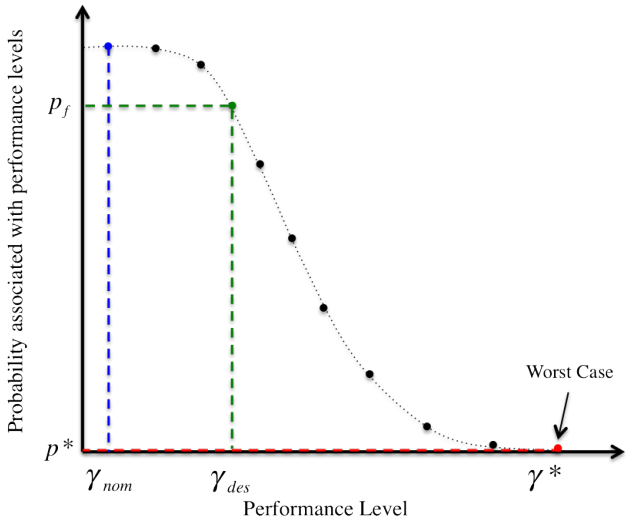


Figure 3.2: The probability profile of performance as a tool for robustness analysis

the failure probability, and expected to be small. Suppose the failure probability is large, two possibilities exist:

- The control design is poor and should be revisited, or
- The desired criterion is selected poorly and should be relaxed. Indeed, relaxation of the performance criterion is done fairly often in the Industry, usually in a naive manner. The PPoP provides a probabilistic rationale for such a relaxation.

The blue shaded area in Figure 3.3 illustrates the "Decay Region", the section of the curve where the probability varies drastically. If γ_{des} lies within this region, there is a benefit in relaxing the criterion, since a small increase in performance leads to a large decrease in the corresponding probability. However, if γ_{des} lies in the green or red regions, the benefit of bound relaxation would be small.

The three regions in the PPoP are defined by taking inspiration from concepts of rise time and settling time in step response curves:

- Accepted Performance Region constitutes performances:corresponding to probabilities from 0.9 to 1.

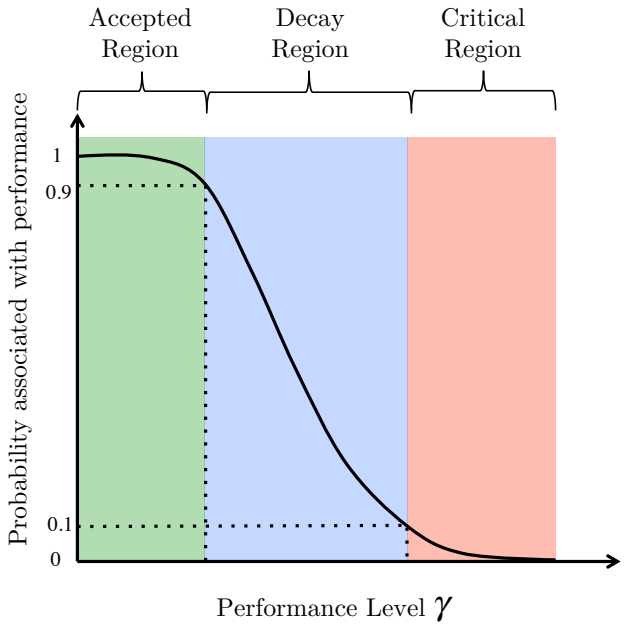


Figure 3.3: Regions in the probability profile of performance

- Decay Region, where the probability varies rapidly. It constitutes performances corresponding to probabilities of 0.1 to 0.9.
- Critical Region: where lower probabilities are encountered, but the performance values are potentially problematic. This region comprises performances corresponding to probabilities lower than 0.1.

Thus, the PPoP curve offers significant insight into a control system's performance [104]. So far, it has been established that computing the probabilities associated with various performance levels is indeed valuable. In the following section, the cross-entropy based methodology is detailed, which will enable the estimation of such probabilities.

3.5 Methodology

This section develops a methodology for estimating the desired probability, first for the general probability distribution, and subsequently for the specific case of the truncated normal distribution. Let the uncertain parameter vector δ be distributed according to a family of probability distribution functions (pdf) $f(\delta; v)$, where v parametrises the family. The probability

$$l := P\left[J\left(\delta\right) \geq \gamma\right] \tag{3.3}$$

relies on the **Indicator Function** $I_{\{J(\delta)\geq\gamma\}}$:

$$I_{\{J(\delta)\geq\gamma\}} = \begin{cases} 1 & \text{if } J\left(\delta\right) \geq \gamma \\ 0 & \text{if } J\left(\delta\right) < \gamma \end{cases}. \tag{3.4}$$

Informally, the events leading to $I_{\{J(\delta)\geq\gamma\}} = 1$ are called **hits**. Estimating the probability in Equation (3.3) requires solving the following NP-Hard integration problem:

$$P\left[J\left(\delta\right) \geq \gamma\right] = E\{I_{\{J(\delta)\geq\gamma\}}\} = \int_{\mathbb{D}} I_{\{J(\delta)\geq\gamma\}} f\left(\delta; v\right) d\delta, \tag{3.5}$$

which essentially amounts to finding the expectation E that $[J(\delta) \geq \gamma]$.

A naive stochastic estimator is Crude Monte Carlo (CMC) simulation [92], in which N samples $\delta_1, \delta_2, ...\delta_i, ...\delta_N$ are drawn from the distribution $f\left(\delta; v\right)$. Recall that each sample δ_i consists of a row vector with k uncertain parameters. Therefore, the samples

may be represented as a matrix with N rows, each consisting of k columns.

$$\delta_{samp} = \begin{bmatrix} \delta_{11}, \ \ldots \delta_{1j}, \ \ldots, \delta_{1k} \\ \delta_{21}, \ \ldots \delta_{2j}, \ \ldots, \delta_{2k} \\ \vdots \\ \delta_{i1}, \ \ldots \delta_{ij}, \ \ldots, \delta_{jk} \\ \vdots \\ \delta_{N1}, \ \ldots \delta_{Nj}, \ \ldots, \delta_{Nk}. \end{bmatrix} \tag{3.6}$$

Specifically, note that the subscript i indexes each sample of the uncertain parameter vector, while the subscript j indexes each particular uncertain parameter within the vector in the mathematical development that follows. Then

$$\hat{l}_{CMC} = \frac{1}{N} \sum_{i=1}^{N} I_{\{J(\delta_i) \geq \gamma\}} \tag{3.7}$$

gives an unbiased probability estimation. This becomes intractable when $\{J(\delta) \geq \gamma\}$ is a rare event i.e., has a small probability of occurrence (say of the order 10^{-6}) [57]. For rare events, an estimator is required that can identify the "important" parameter regions that cause the performance value to reach extreme values. Importance sampling (IS) techniques are a promising class of sampling techniques which efficiently bias the input distributions [26]. For rare events, it is found that the cross entropy (CE) simulation technique is particularly suited [92]. This technique is described in the following section.

3.6 The Cross Entropy Method for Rare Event Simulation

CE is an adaptive importance sampling algorithm, developed in [90] to determine the probabilities of rare events. CE adaptively computes the best reference parameter(s), and biases the initial uncertain parameter distribution so that the rare event becomes

more likely to occur in the new distribution [92]. Subsequently, an importance sampling run computes the probability of occurrence of the rare event by scaling back from the new distribution to the original distribution, with the help of a likelihood ratio. The principle of CE is illustrated in Figure 3.4.

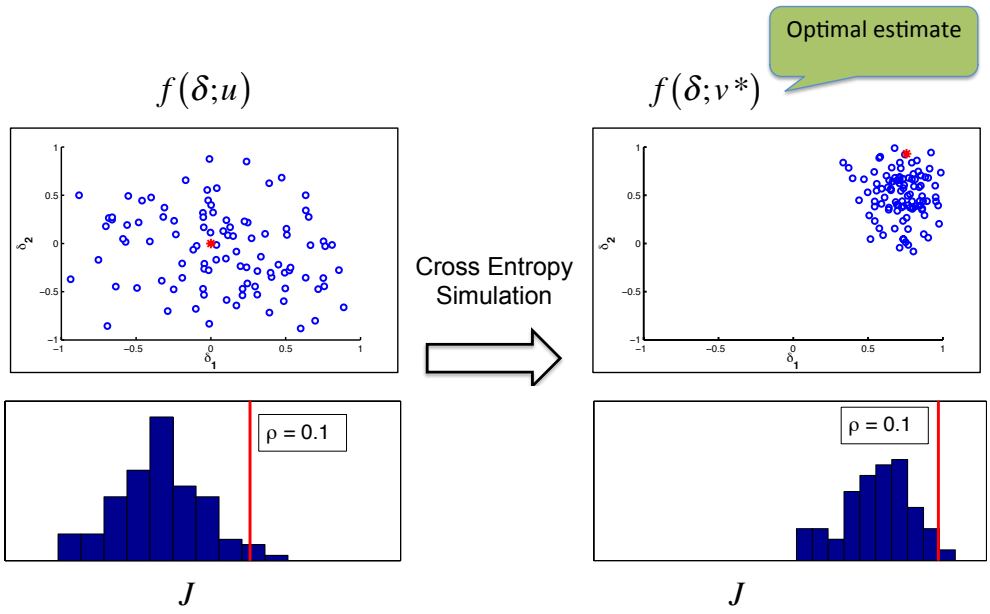


Figure 3.4: Illustration of CE principle. The simulation takes an unbiased input distribution, and biases it towards regions of the parameter space that cause poor performance by finding an optimal reference parameter.

Importance sampling seeks to find an *importance sampling density* or the *instrumental density* [26]. Let $g(\delta)$ defined on $\mathbb{D}$ be this unknown distribution, to be found. Suppose that N random samples $\delta_1, ..., \delta_N$ are drawn from $g(\delta)$. Let the initial distribution be $f(\delta; u)$, where u is the initial reference parameter. A reference parameter refers to any parameter that defines the behaviour of the distribution. For instance, the upper and lower bounds constitute the parameters in a uniform distribution, while the mean and standard deviation constitute the parameters in a normal distribution. In this work, reference parameters are considered as vectors in the interest of generality.

l may be found by modifying equation (3.5) as follows:

$$E\{J(\delta) \geq \gamma\} = \int_{\mathbb{D}} I_{\{J(\delta)\geq\gamma\}} \frac{f(\delta;u)}{g(\delta)} g(\delta)\, d\delta \tag{3.8}$$

The stochastic estimator of this is written, similar to (3.7), as follows:

$$\hat{l} = \frac{1}{N} \sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} \frac{f(\delta_i;u)}{g(\delta_i)}. \tag{3.9}$$

The optimal estimate is then found by having g^* satisfying:

$$g^*(\delta) := \frac{I_{\{J(\delta)\geq\gamma\}} f(\delta;u)}{l} \tag{3.10}$$

which depends on l and so is unknown. The manner in which $g(.)$ is chosen varies across different IS methods. The particular choice of $g(.)$ in CE is detailed in the following section.

3.6.1 Estimation of Rare Event Probabilities

CE is particularly effective at estimating rare-event probabilities, i.e., events whose probabilities of occurrence are smaller than 10^{-6}. CE chooses g from within the same family of densities $f(\cdot, v)$, with an "optimal" reference parameter v, which is chosen such that some measure of *distance* between the densities $g^*(.)$ and $f(\cdot;v)$ is minimum. In the CE method, the Kullback-Leibler divergence, also known as the cross entropy, is considered to be the distance between the distributions. For $g^*(\delta)$ and $f(\delta;v)$, the Kullback-Leibler distance is:

$$\begin{aligned} \mathcal{D}(g^*(\delta), f(\delta;v)) = &\int g^*(\delta) \ln(g^*(\delta))\, d\delta \\ &- \int g^*(\delta) \ln(f(\delta;v))\, d\delta \end{aligned} \tag{3.11}$$

Minimising the Kullback-Leibler divergence for $g^*(.)$ and $f(\delta; v)$ is equivalent to minimising the second term $-\int g^*(\delta) \ln(f(\delta; v))\, d\delta$, which is in turn equivalent to solving the maximisation problem:

$$\max_v \int g^*(\delta) \ln(f(\delta; v))\, d\delta. \tag{3.12}$$

Substituting for $g^*(.)$ from (3.10), the optimisation program becomes

$$\max_v \int \frac{I_{\{J(\delta)\geq\gamma\}} f(\delta; u)}{l} \ln(f(\delta; v))\, d\delta. \tag{3.13}$$

If the likelihood ratio W is defined as

$$W(\delta; u, v) = \frac{f(\delta; u)}{f(\delta; v)} \tag{3.14}$$

the solution to the optimisation program may be found by solving for the optimal updating factor v^* in the following equation:

$$\frac{1}{N} \sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} W(\delta_i; u, v) \nabla \ln(f(\delta_i; v^*)) = 0. \tag{3.15}$$

For the normal distribution, $v = \begin{bmatrix} \mu & \sigma \end{bmatrix}$ and $f(\delta; v) = f\left(\delta; \begin{bmatrix} \mu & \sigma \end{bmatrix}\right)$

$$\int I_{\{J(\delta)\geq\gamma\}} W\left(\delta; \begin{bmatrix} \mu_0 & \sigma_0 \end{bmatrix}, \begin{bmatrix} \mu & \sigma \end{bmatrix}\right) f\left(\delta; \begin{bmatrix} \mu & \sigma \end{bmatrix}\right) \begin{bmatrix} \frac{\partial}{\partial \mu} \ln\left(f\left(\delta; \begin{bmatrix} \mu & \sigma \end{bmatrix}\right)\right) \\ \frac{\partial}{\partial \sigma} \ln\left(f\left(\delta; \begin{bmatrix} \mu & \sigma \end{bmatrix}\right)\right) \end{bmatrix} d\delta = 0. \tag{3.16}$$

Substituting $f\left(\delta;\begin{bmatrix}\mu & \sigma\end{bmatrix}\right) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(\delta-\mu)^2}{2\sigma^2}}$ in (3.16) and separating the components:

$$\int I_{\{J(\delta)\geq\gamma\}}W\left(\delta;\begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix},\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)f\left(\delta;\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)\frac{\partial}{\partial\mu}\left(\ln\left(\frac{1}{\sigma\sqrt{(2\pi)}}\right)-\frac{(\delta-\mu)^2}{2\sigma^2}\right)d\delta = 0. \tag{3.17}$$

$$\int I_{\{J(\delta)\geq\gamma\}}W\left(\delta;\begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix},\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)f\left(\delta;\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)\frac{\partial}{\partial\sigma}\left(\ln\left(\frac{1}{\sigma\sqrt{(2\pi)}}\right)-\frac{(\delta-\mu)^2}{2\sigma^2}\right)d\delta = 0. \tag{3.18}$$

The updating rule derivation for the mean μ from 3.17 is now described. First, the logarithmic term is only expressed in terms of σ, and therefore the partial derivative in terms of μ is reduced to 0.

$$\int I_{\{J(\delta)\geq\gamma\}}W\left(\delta;\begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix},\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)f\left(\delta;\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)\left(0-\frac{\partial}{\partial\mu}\frac{(\delta-\mu)^2}{2\sigma^2}\right)d\delta = 0. \tag{3.19}$$

Now, carrying out the partial derivative $\frac{\partial}{\partial\mu}\frac{(\delta-\mu)^2}{2\sigma^2}d\delta$ gives

$$\int I_{\{J(\delta)\geq\gamma\}}W\left(\delta;\begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix},\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)f\left(\delta;\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)\left(\frac{2\delta-2\mu}{2\sigma^2}\right)d\delta = 0. \tag{3.20}$$

The left side of the above equation may be written as the expectation $E_v\{I_{\{J(\delta)\geq\gamma\}}W\left(\delta;\begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix},\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)(\delta-\mu)\}$, where the subscript v indicates that it is taken with respect to the density $f(\delta;v) = f\left(\delta;\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)$. This is expressed stochastically as follows:

$$\frac{1}{N}\sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}}W\left(\delta_i;\begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix},\begin{bmatrix}\mu & \sigma\end{bmatrix}\right)(\delta_i-\mu) = 0 \tag{3.21}$$

From the above equation, the updating rule for the mean may be found as follows:

$$\mu^* = \frac{\sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} W\left(\delta_i; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)\delta_i}{\sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} W\left(\delta_i; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)} \tag{3.22}$$

The updating rule for the standard deviation is now derived from (3.18):

$$\int I_{\{J(\delta)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) f\left(\delta; \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) \frac{\partial}{\partial\sigma}\left(\ln\left(\frac{1}{\sqrt{2\pi}}\right) + \ln\left(\frac{1}{\sigma}\right) - \frac{(\delta-\mu)^2}{2\sigma^2}\right) d\delta = 0. \tag{3.23}$$

i.e.,

$$\int I_{\{J(\delta)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) f\left(\delta; \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) \left(\left(\frac{-1}{\sigma}\right) + \frac{(\delta-\mu)^2}{\sigma^3}\right) d\delta = 0. \tag{3.24}$$

Separating the additive terms:

$$\begin{aligned}\int I_{\{J(\delta)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) f\left(\delta; \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) \left(\frac{-1}{\sigma}\right) d\delta = \\ \int I_{\{J(\delta)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) f\left(\delta; \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) \left(\frac{(\delta-\mu)^2}{\sigma^3}\right) d\delta.\end{aligned} \tag{3.25}$$

This gives:

$$\sigma^2 = \frac{\int I_{\{J(\delta)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) f\left(\delta; \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) (\delta-\mu)^2}{\int I_{\{J(\delta)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right) f\left(\delta; \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)} \tag{3.26}$$

Drawing N samples of δ from $f\left(\delta; \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)$, the stochastic estimator of the above is

arrived at, as follows:

$$\sigma^{*2} = \frac{\frac{1}{N}\sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)(\delta_i - \mu)^2}{\frac{1}{N}\sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)} \tag{3.27}$$

This gives the updating rule for the standard deviation as:

$$\sigma^{*} = \sqrt{\frac{\sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)(\delta_i - \mu)^2}{\sum_{i=1}^{N} I_{\{J(\delta_i)\geq\gamma\}} W\left(\delta; \begin{bmatrix}\mu_0 & \sigma_0\end{bmatrix}, \begin{bmatrix}\mu & \sigma\end{bmatrix}\right)}} \tag{3.28}$$

where $i = 1, ..., N$, $j = 1, ..., k$. Here, $\delta_{i,j}$ represents the j^{th} element of the i^{th} sample δ_i (recall that each sample of the uncertain parameter belongs to $\mathbb{R}^k$).

The updating rules for the mean and standard deviation are now included in an algorithm to iteratively optimise the parameter distributions so that they yield high performance values when simulated. Such an algorithm is developed in the following section.

3.6.2 The CE Algorithm

As discussed, the objective is to adaptively identify narrow parameter distributions (by computing the best reference parameters, μ^* and σ^*) such that the KL distance is minimum. This is carried out in a multi-stage algorithm by finding the sequence of reference parameters $\{[\mu_t\ \sigma_t], t \geq 0\}$ and a corresponding sequence of performance levels $\{\gamma_t, t \geq 1\}$ and iterating until the quantile of performance $\gamma_t \geq \gamma$. Details of the generalised algorithm may be found in [57]. The algorithm biases the initial distribution towards the region in the parameter space that causes higher performances to occur. Then, this distribution is scaled back using the likelihood ratio to determine

the probability of occurrence of the performance value in the original performance.

The CE algorithm is shown in Algorithm 1. This is carried out for each performance level of interest γ. Here, N is the number of evaluations per CE iteration t, ρ is the rarity factor, $maxits$ represents the maximum number of CE iterations allowed. N_1 is the number of evaluations for the final importance sampling run. Generally speaking the choice of N and N_1 rely on the model under study, and on the computational budget. In any case both of these are far lower than for naive Monte Carlo. ρ, the rarity factor is the probability below which a CE updating iteration is carried out. In our study, ρ is usually set at 0.1, corresponding to the decay region in Figure 3.3. σ^* is the threshold standard deviation below which CE updating is terminated. Therefore, three termination criteria exist at the t^{th} iteration:

- the performance quantile $\gamma_t \geq \gamma$, or
- $min(\sigma(t+1)) \leq \sigma^*$, or
- $t > maxits$

Generally speaking, the parameter distribution f, the mean μ_0 and standard deviation σ_0 can be the same for a specific application[1]. All other input parameters to the algorithm are based on computational resource considerations. Therefore, the external "appearance" of CE is similar to that of MC to a system analyst. In other words, the analyst need only decide the number of samples allowed, by taking into account the computational budget. The uncertain parameter vector samples are created by CE, and the output performance values J are output, which can be interpreted post-analysis. This MC-like appearance is attractive because it implies that CE can straightforwardly be implemented in existing industry $V\&V$ frameworks, most of which are MC-based.

[1] for instance, in the analysis of launch vehicles, the initial distribution of fuel mass may vary with a Gaussian distribution with a normalised mean of 0 and a normalised standard deviation of 0.4

Data: μ_0,σ_0, $maxits$, ρ, σ^*, N and N_1;
Result: The rare-event estimate $\hat{l}$;
Set $t = 1$;
Set $\mu(t) = \mu_0$ and $\sigma(t) = \sigma_0$;
while $t \leq maxits$ **do**
1. Generate a sample $\delta_1, ..., \delta_N$ from the density $f(.; \mu(t), \sigma(t))$;
2. Compute γ_t, the sample $(1-\rho)$ performance quantile as follows:
   2a. Calculate the values of the cost $J(\delta_i)$ for each sample δ_i.;
   2b. Arrange them in ascending order. Then find $\gamma_t = J_{[(1-\rho)]}$.;
3. Use the same sample $\delta_1, ..., \delta_N$ to find the updating parameter as follows:

$$\mu_j(t+1) = \frac{\sum_{i=1}^{N} I_{\{J(\delta_{i,j})\geq\gamma\}} W(\delta_{i,j}; [\mu_0, \sigma_0], [\mu(t), \sigma(t)])\, \delta_{i,j}}{\sum_{i=1}^{N} I_{\{J(\delta_{i,j})\geq\gamma\}} W(\delta_{i,j}; [\mu_0, \sigma_0], [\mu(t), \sigma(t)])}$$

and

$$\sigma_j(t+1) = \sqrt{\frac{\sum_{i=1}^{N} I_{\{J(\delta_{i,j})\geq\gamma\}} W(\delta_{i,j}; [\mu_0, \sigma_0], [\mu(t), \sigma(t)])\, (\delta_{i,j} - \mu_j(t+1))^2}{\sum_{i=1}^{N} I_{\{J(\delta_{i,j})\geq\gamma\}} W(\delta_{i,j}; [\mu_0, \sigma_0], [\mu(t), \sigma(t)])}}$$

if $min(\sigma(t+1)) \leq \sigma^*$ **then**
  $\hat{\mu}_T = \hat{\mu}_(t+1)$;
  $\hat{\sigma}_T = \hat{\sigma}_(t+1)$;
  Proceed to step 4
end
if $\gamma_t \geq \gamma$ **then**
  $\gamma_t = \gamma$;
  $\hat{\mu}_T = \hat{\mu}_t$;
  $\hat{\sigma}_T = \hat{\sigma}_t$;
  Optimal reference parameter found, proceed to step 4;
else
  $t = t + 1$;
end
end
4. Estimate the rare-event probability l using the likelihood ratio estimator:

$$\hat{l} = \frac{1}{N_1} \sum_{i=1}^{N_1} I_{\{J(\delta)\geq\gamma\}} W(\delta_{i,j}; [\mu_0, \sigma_0], [\mu_T, \sigma_T]);$$

Algorithm 1: The CE Algorithm for Rare Event Simulation

3.7 Generating the PPoP

In order to generate the PPoP curve, $\hat{l} = P\left[J\left(\delta\right) \geq \gamma\right]$ is to be computed at various levels of γ, and a curve fitting is done. Generally speaking, a good starting performance level is the nominal performance γ_{nom}, i.e., the value of performance evaluated at $\delta = [0, \ldots, 0]^T$. The rationale for this is that the performance is expected to always degrade in the presence of uncertainty, and not improve. Since it is expected that the worst case γ^* is a rare-event, the stopping level is γ^*, or the next nearest decimal point depending on the step-size chosen. The step-size $\Delta\gamma$ is a design parameter, which specifies how fine the gridding is, and hence, how accurate the curve fitting is for the probability profile.

Then, a vector Γ of performance levels is specified as $\Gamma = [\gamma_{nom}, \gamma_{nom} + \Delta\gamma, \ldots, \gamma^*]$.

Then, the algorithm is:

for $ii = 1 : length(\Gamma)$

$[\hat{l}(ii), \mu_T(ii, :), \sigma_T] = call\_CE(\Gamma(ii))$;

end

$plot(\Gamma(ii), \hat{(l)})$

3.7.1 Adaptive Initialisation

An adaptive initialisation mechanism is introduced in the algorithm to increase its efficiency further, where the initial distribution $f(\delta; \mu_0, \sigma_0)$ is adaptively varied. This scheme takes advantage of the computations already carried out. The initial distribution is biased towards the mean and standard deviations already found, i.e.,

$$\mu_0(ii + 1) = \mu_T(ii)$$

Therefore, the search space is shifted towards the region in the parameter space that already leads to higher costs (or worse performance). The standard deviation is kept

the same as that of the initial distribution (i.e. 0.4) so as to not excessively restrict the size of the search space. Such a scheme is illustrated in Figure 3.5

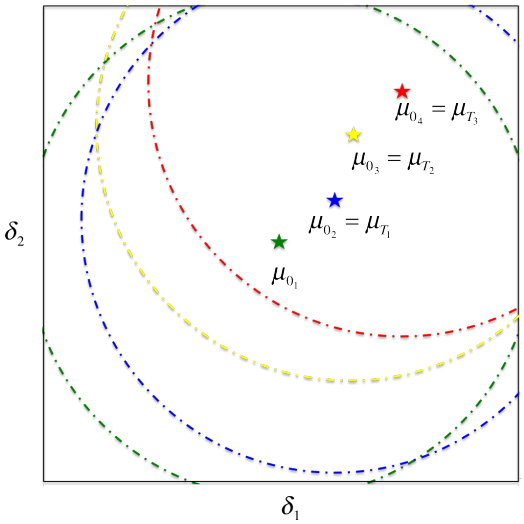


Figure 3.5: CE adaptive initialisation scheme

3.8 Illustrative Example

In this section, the algorithm developed in this chapter is illustrated using a simple example [105]. Consider the following linear uncertain system

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} X + \begin{bmatrix} 0 \\ 1 \end{bmatrix} U + \begin{bmatrix} 0 \\ 1 \end{bmatrix} W_d \tag{3.29}$$

$$Z = \begin{bmatrix} 1 & 0 \end{bmatrix} X \tag{3.30}$$

with parameters $a_0 = \overline{a}_0 + q_0$ and $a_1 = \overline{a}_1 + q_1$, where the nominal values of the parameters are $\overline{a}_0 = 1$ and $\overline{a}_1 = 0.8$, and the uncertain parameters $q = \begin{bmatrix} q_0 & q_1 \end{bmatrix}^T$ are in

the set $\mathbb{Q} = \{q \in \mathbb{R}^2 : q_0 \in [-0.025, 0.025], q_1 \in [-0.025, 0.025]\}$. These uncertainties are expressed as $q_0 = q_{0_{nom}} + 0.025 \times \delta_{q_0}$ and $q_1 = q_{1_{nom}} + 0.025 \times \delta_{q_1}$, where $q_{0_{nom}}$ = $q_{1_{nom}}$=1. W_d are measured inputs and Z are measured outputs. The performance criterion under consideration is the $\mathcal{H}_\infty$ norm of the uncertain system. The worst case is found in [105] to be 1.4, and is achieved when $q_0 = q_1$ =-0.025. Figure 3.6 shows the Bode plot for the system, which illustrates that the maximum $\mathcal{H}_\infty$ norm obtained, i.e., the outer envelope, is at 1.4.

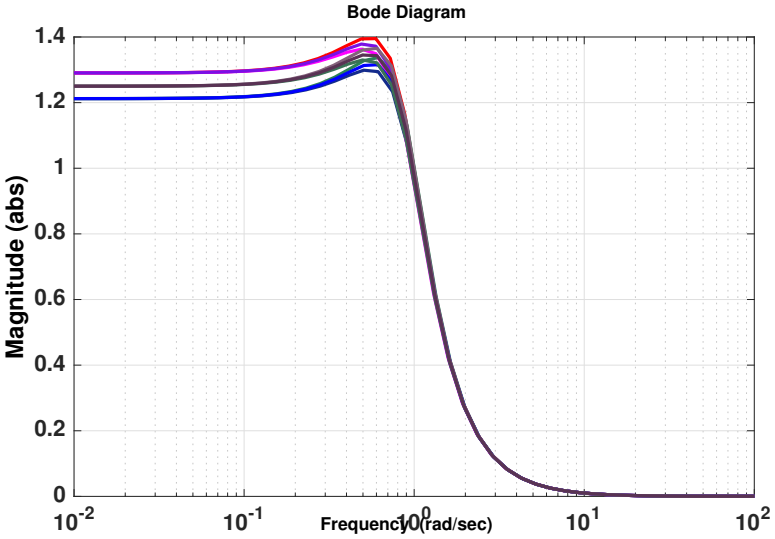


Figure 3.6: Bode plot showing gain of the illustrative system.

The CE algorithm is applied to this system to estimate the probability associated with the occurrences of various performance levels. The simulation settings used are $N = 100$, $N_1 = 1000$, $\rho = 0.01$ and initial mean value of $u = [0, 0]$. These results are shown in Figure 3.7. It is apparent that the probability of the system with performance values greater than 1.4 dB is 0. An interesting observation is made at $\gamma = 1.397$ dB, at which the probability of occurrence is very low, of the order of 10^{-7}. This shows the efficacy of cross entropy for finding the probability of occurrence of rare-events, and

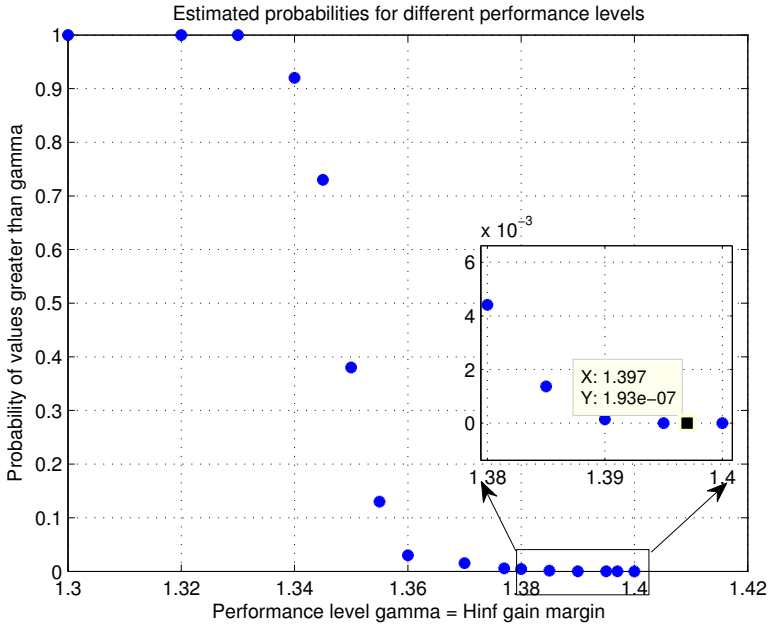


Figure 3.7: Plot of probabilities of occurrence of increasing performance levels of $\mathcal{H}_\infty$ norm of the illustrative system.

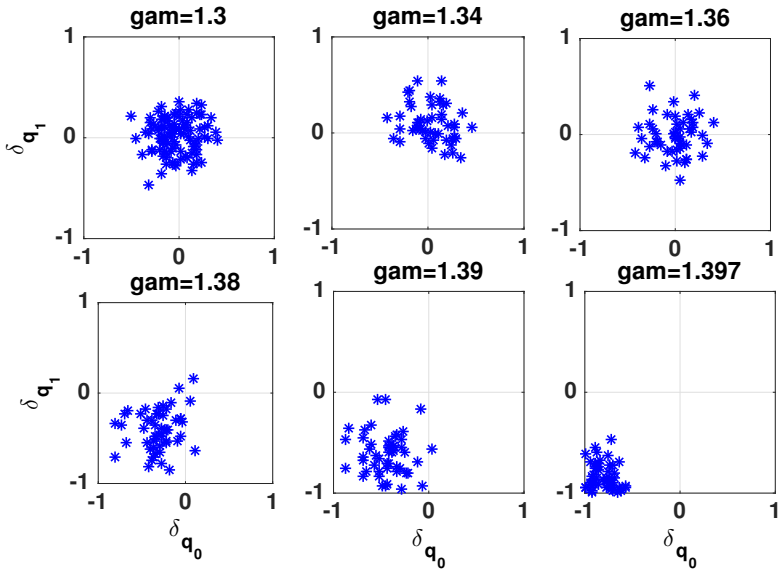


Figure 3.8: Variation of the uncertain parameter samples at the final IS run as the performance level varies.

for finding the probability of occurrence of worst-case values. Further, the probability profile at various performance levels is also found by the algorithm. Observe that there is significant degradation in performance between the levels 1.33 dB and 1.38 dB. Further, the probability associated with performances greater than 1.33 dB is 1, which implies that the peak performance is guaranteed to be above 1.33 dB at all parameter values.

Suppose, for the sake of illustration, that the performance criterion is that the $\mathcal{H}_\infty$ norm is to remain less than 1.395 units. Clearly, the classical robust performance (or deterministic robust performance) is not satisfied in this case. However, the probability associated with γ, i.e., $P[J \geq 1.395] = 3.2 \times 10^{-6}$, which is low. If the performance criterion is instead defined probabilistically, for example, "the probability that the performance criterion exceeds 1.395 is lower than 10^{-5}". This is simply a restatement of the probabilistic robust performance criterion in Equation (2.23), with $\gamma^* = 1.395$ $p_{critical} = 10^{-5}$. It may be found that the system in Equations (3.29) and (3.30) meet this criterion. Thus, this system does not satisfy deterministic robust performance, but does satisfy probabilistic robust performance. Such notions are valuable in practical industry systems, as will be seen in the results in later chapters.

The samples drawn for the IS run carried out at the final iteration are shown for different values of γ in Figure 3.8. It is clear that the algorithm causes the mean parameter v to converge towards the region $[-1, -1]$ as performance degrades. Observe that for $\gamma = 1.3$ dB, the algorithm does not change the initial distribution since the occurrence of the performance level of γ =1.3 is frequent enough. At performances closer to the worst case value of $\gamma = 1.4$, the samples converge towards the problematic [-1,1] region. The samples drawn in the final IS runs over all γ levels, and their corresponding performance values are scattered to form a heat map representation of the uncertain

parameter region, as shown in Figure 3.9. The colour intensities are mapped to the performance values, thereby creating a plot reminiscent of safe and unsafe regions [18].

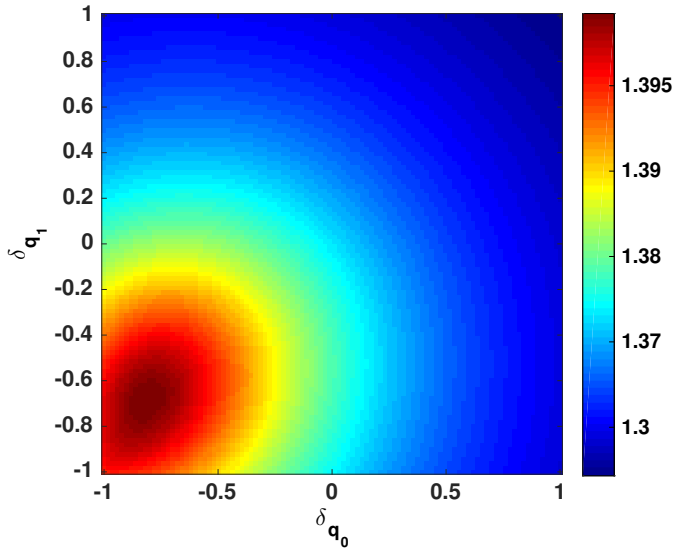


Figure 3.9: Heat map showing uncertain parameter regions corresponding to various performance values

3.9 Conclusions

This chapter proposes the cross entropy-based simulation technique for estimation of rare event probabilities. The concept of the probability profile of performance is introduced, and an algorithm based on CE-based methods is developed to generate it. A simple illustrative example is then presented to demonstrate these concepts. The algorithm developed in this chapter will be employed in Chapters 5 and 6. The next chapter details the development of models for autonomous rendezvous of the Mars Sample Return mission, required for the analysis detailed in Chapters 5.

Chapter 4

Modelling the Autonomous Rendezvous System

4.1 Background

This chapter is focussed on the modelling of autonomous rendezvous in the Mars Sample Return mission. This research activity was undertaken as part of a European Space Agency project, carried out in collaboration with the aerospace companies GMV, Spain, and Thales Alenia Space France (TASF)[1]. The project is the Autonomous Integrated Robust GNC (iGNC[2]) for the Mars Sample Return (MSR) mission [106]. The MSR is an ESA mission to be carried out in the 2020s, with the objective of collecting Martian soil and rock samples and retrieving them to Earth for scientific research. The results of the study can potentially answer questions about the possibility of life on Mars, in the present or the past. The MSR is a complex multi-spacecraft mission that has raised several engineering challenges. The mission starts with a Mars lander launched from Earth, which contains the sample container. This container is filled with the sample

[1] ESA, GMV, TASF and University of Exeter form the iGNC project consortium.

[2] Although the formal abbreviation is AIRGNC, it has been shortened in publications for brevity's sake.

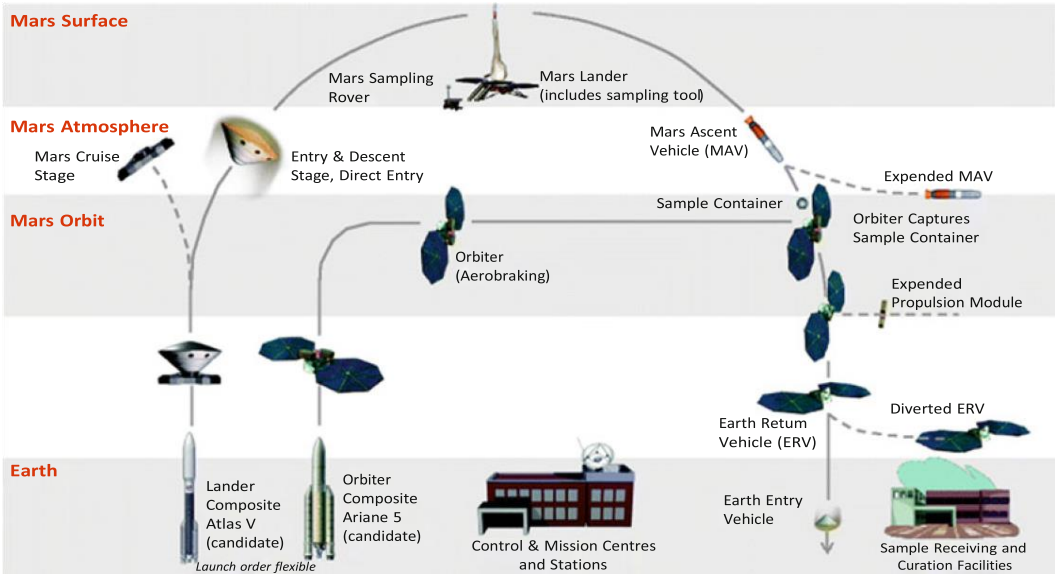


Figure 4.1: Mars Sample Return mission scenario. Source: iMARS

upon landing on Mars, and then launched by the Mars Ascent Vehicle (MAV) to Mars orbit. An Earth Return Vehicle (ERV) launched from Earth then collects the sampling container by performing a rendezvous manoeuvre in deep space. The ERV returns to Earth with the container, after which the sample is sent to special facilities for decontamination and the start of scientific study. A mission scenario conceived by iMARs [107] is shown in Figure 4.1. The stage of the mission focused on in this study is the rendezvous and docking of the ERV and the Orbiting Sample (OS). It has been documented [108] that this sub-mission is complicated, due to the autonomous manoeuvres and safety considerations, showing that its $V\&V$ is a critical step contributing to success of the entire MSR mission.

The project activities dealt with in this study are (1) to develop LFT formulations of linear/linearised models for control design, and (2) to conduct a performance analysis of the designed controller. These are, therefore, the central focus of this chapter and the next. These activities complement other studies pertaining to control design conducted

by the project consortium, and fit into the frame of the project. First, a brief overview of the model is presented. Two levels of modelling are distinguished. The first is a simplified rendezvous model necessary for robust controller synthesis and design stage analysis. The next is a full high-fidelity model used for final stage analysis in the presence of the controller and all possible behaviours and uncertainties included.

4.2 Model Overview

The Finite Element Simulation (FES) real world architecture in Figure 4.2 shows the systems involved in the rendezvous and docking manoeuvres between the chaser spacecraft (in this case, the chaser is the ERV) and the target (the orbiting sample). The architecture constitutes the models that must be developed to accurately depict the behaviours of all systems involved [109]. It is also the environment in which the synthesized GNC algorithm is to be tested. It accounts for the dynamics and uncertainties across all the systems. It includes the chaser spacecraft (SC) systems, orbital mechanics, communication links to various locations (Earth control centre, Deimos, Phobos and Mars), telemetry and telecommunication, and camera visibility. It also includes the MAV environment and OS seperation dynamics, since any uncertainty in the OS launch window will affect the rendezvous trajectory. The chaser actuators are significant sources of uncertainty, particularly misalignment angles in the thrusters. Several of these models are available internally in the libraries of ESA and GMV. In this chapter, the models of some relevant chaser spacecraft systems is described, which are important for controller design and analysis[1]. The OS model is typically straightforward, and is modelled by the 6DOF dynamics of a rigid body [109]. In the study it is assumed that OS orbital insertion is done successfully by the MAV, and only the chaser SC controller carries out the rendezvous manoeuvres. This is consistent with ESA's mission scenario.

[1] All of the rendezvous GNC maneuvers are performed by the chaser, while the OS GNC is limited to orbiting Mars after its launch by the MAV

4.2.1 Frames of Reference

Before proceeding to describe the models, it is important to list the frames of reference, particularly important with the involvement of multiple spacecraft. Since rendezvous in general involves complex relative manoeuvres, various frames of reference are considered — a comprehensive list is found in [110]. The following frames of reference are relevant to this study.

- $\mathcal{F}_i$: The inertial frame $\mathcal{F}_i$ has its origin at the centre of earth, with the x-axis $\mathcal{X}_i$ extending from the origin along the vernal equinox line. The z-axis $\mathcal{Z}_i$ is to the north along the angular momentum vector of the Earth. Then, $\mathcal{Y}_i = \mathcal{Z}_i \times \mathcal{X}_i$ completes the triad.

- $\mathcal{F}_o$: The local orbital frame $\mathcal{F}_o$ has its origin at the centre of mass of the chaser SC, with the x-axis $\mathcal{X}_o$ along the direction of velocity vector of the spacecraft. In rendezvous literature this direction is also known as $V - bar$. The z-axis $\mathcal{Z}_o$ is in the orbital plane from the spacecraft centre of mass towards the Earth centre. This is also known as $R - bar$. Then, $\mathcal{Y}_o = \mathcal{Z}_o \times \mathcal{X}_o$, and is normal to the orbital plane and in the opposite direction and parallel to the orbital angular momentum vector. $\mathcal{Y}_o$ is referred to as $H - bar$. $\mathcal{F}_o$ is known also as the Local Vertical Local Horizontal (LVLH) frame [111].

- $\mathcal{F}_{bc}$: The chaser body reference frame. $\mathcal{F}_{bc}$ has its origin at the chaser centre of mass, and the X-axis $\mathcal{X}_{bc}$ is the longitudinal direction towards the docking port and normal to the A0 ring plane. $\mathcal{Y}_{bc}$ is along the geometrical centre line of the solar panels in the A0 plane, and $\mathcal{Z}_{bc} = \mathcal{X}_{bc} \times \mathcal{Y}_{bc}$ [9].

- $\mathcal{F}_{bt}$: The target body reference frame, with origin at the OS centre of mass. $\mathcal{X}_{bt}$ is along the direction of the orbiting sample's velocity vector. $\mathcal{Y}_{bt}$ is perpendicular to the surface of the orbiting sample cylinder, and $\mathcal{Z}_{bt} = \mathcal{Y}_{bt} \times \mathcal{X}_{bt}$.

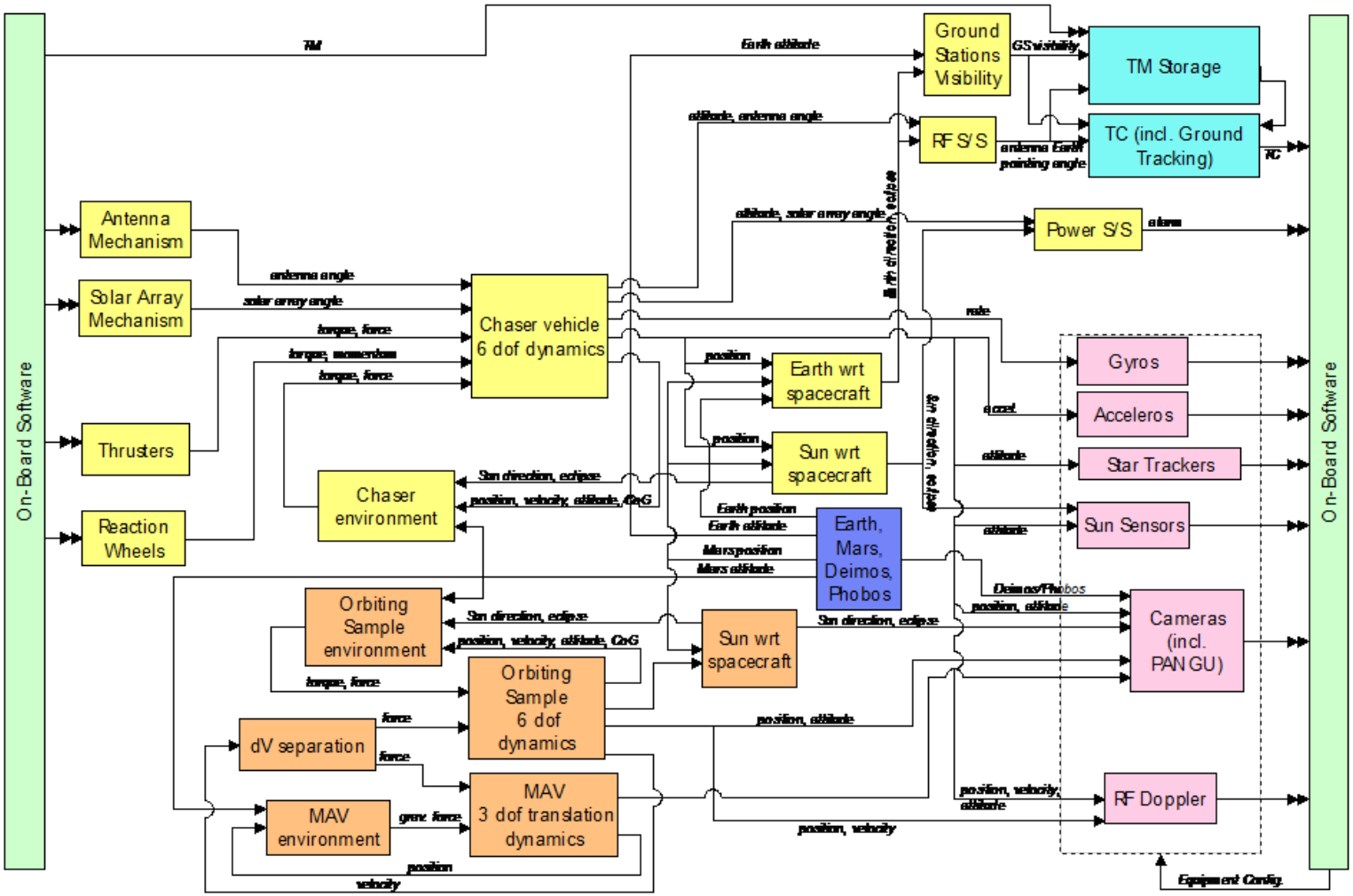


Figure 4.2: FES real world architecture of the iGNC mission

4.3 Simplified Rendezvous Model

The simplified rendezvous model refers to a set of interconnected models used in the robust controller development. Classically, it usually refers to the linear system environment. There exist also simplifying assumptions in the models, such as neglecting sensor bias values or ignoring higher frequency modes. It includes linearised systems and a smaller uncertainty subset, mainly consisting of chaser dynamics, actuators and sensors. It does not include MAV dynamics or orbital mechanics. The control design and initial analysis is carried out in this environment. For the purposes of LFT modelling, some of the models are presented in this chapter.

4.3.1 LFT Modelling

LFT models of the chaser spacecraft (the ERV) are vital for designing a robust controller for the Rendezvous and Docking (RvD) mission. Developing LFT formulations for a subset of the chaser SC models was Exeter's responsibility. Some basics of LFT modelling are reviewed in the Chapter 2. The LFTs are to be developed for linear and linearised systems, particularly the dynamics including flexible mode and sloshing effects, sensors and actuators. This section describes the modelling of one of the sensors, i.e. the accelerometer. This system is chosen for the purposes of illustration as it is the most straightforward, and will allow the steps to be discussed with clarity.

4.3.2 LFT Modelling of Linearised Accelerometer Model

The chaser spacecraft's linearised accelerometer is chosen as an example for LFT modelling. The accelerometer is responsible for measuring the spacecraft linear acceleration. It provides the measurement in the accelerometer axes, which must be transposed to the chaser body frame, in order for the measured value to be incorporated in calculating the controller inputs. Two accelerometers are included in the chaser spacecraft. Their

model is described as follows:

$$Y_A = [D_A] \times \alpha_{bc} \tag{4.1}$$

where Y_A is the measured accelerometer deviation vector from the reference chaser acceleration. α_{bc} is the chaser acceleration deviation from the reference value expressed in the accelerometer frame. D_A is the accelerometer direction matrix composed of the components of the accelerometer measurement axes expressed in the $\mathcal{F}_{bc}$ frame.

A rotation needs to be carried out to compute the acceleration in the $\mathcal{F}_{bc}$ frame i.e.,

$$D_A = R_A * D'_A \tag{4.2}$$

where R_A is the rotation matrix composed of the uncertain misalignment angles $\delta_{A_x}, \delta_{A_y}, \delta_{A_z}$ in the $x-, y-$, and $z-$ axes respectively. For small angle misalignments [112], the rotation matrix may be approximated to:

$$R_A = \begin{bmatrix} 1 & \delta_{A_z} & -\delta_{A_y} \\ -\delta_{A_z} & 1 & \delta_{A_x} \\ \delta_{A_y} & -\delta_{A_x} & 1 \end{bmatrix}, \tag{4.3}$$

The direction matrix D'_A of the accelerometer is assumed to be constant. The matrix is written in the general case as follows:

$$D'_A = \begin{bmatrix} k_{A_1} & k_{A_2} & k_{A_3} \\ k_{A_4} & k_{A_5} & k_{A_6} \\ k_{A_7} & k_{A_8} & k_{A_9} \end{bmatrix}. \tag{4.4}$$

The constants k_{A_1}-k_{A_9} depend on the specific arrangement of the accelerometer w.r.t

the spacecraft $\mathcal{F}_{bc}$ frame. The inputs α_{bc} and outputs are represented as $U_{A_1}, U_{A_2}, U_{A_3}$ and $Y_{A_1}, Y_{A_2}, Y_{A_3}$ respectively, in the interest of generality. Generalising the variables allows easy substitution for similar models. In this study, all of the linearised sensor and actuator systems have similar models, with the measured signal being transformed via a misalignment rotation matrix to yield the "true" values. Upon substituting in Equation (4.2) and multiplying, the equations are expressed as follows:

$$\begin{bmatrix} Y_{A_1} \\ Y_{A_2} \\ Y_{A_3} \end{bmatrix} = \begin{bmatrix} k_{A_1} - k_{A_2}\delta_{A_z} + k_{A_3}\delta_{A_y} & k_{A_1}\delta_{A_z} + k_{A_2} - k_{A_3}\delta_{A_x} & -k_{A_1}\delta_{A_y} + k_{A_2}\delta_{A_x} + k_{A_3} \\ k_{A_4} - k_{A_5}\delta_{A_z} + k_{A_6}\delta_{A_y} & k_{A_4}\delta_{A_z} + k_{A_5} - k_{A_6}\delta_{A_x} & -k_{A_4}\delta_{A_y} + k_{A_5}\delta_{A_x} + k_{A_6} \\ k_{A_7} - k_{A_8}\delta_{A_z} + k_{A_9}\delta_{A_y} & k_{A_7}\delta_{A_z} + k_{A_8} - k_{A_9}\delta_{A_x} & -k_{A_7}\delta_{A_y} + k_{A_8}\delta_{A_x} + k_{A_9} \end{bmatrix} \begin{bmatrix} U_{A_1} \\ U_{A_2} \\ U_{A_3} \end{bmatrix} \tag{4.5}$$

In order to pull out the uncertainties $\delta_{A_x}, \delta_{A_y}, \delta_{A_z}$, the internal inputs and outputs W and Z must be selected. For this purpose, a block diagram interconnection of Equation (4.5) is drawn and examined, as shown in Figure 4.3. In drawing the block diagram, care must be taken not to introduce redundancy in the uncertain parameter occurrence.

For example, consider the first occurrence of δ_{A_y} in Figure 4.3. Two signals $U_1 k_{A_3}\delta_{A_y}$ and $U_1 k_{A_6}\delta_{A_y}$ are needed in different adders. This could prompt two separate δ_{A_y} blocks, such as in Figure 4.4a. However, this leads to two occurrences of δ_{A_y} simply on account of different constant multiplicands. Instead, the interconnection may be modified as in Figure 4.4a by scaling the same uncertain block output to represent two different signals. As a rule of thumb, if a linear combination between signals exists, this needs to be taken advantage of in order to avoid redundant repetitions of uncertainty occurrence. Such basic manipulation prevents the size of the Δ matrix from being excessively large. Recall that large dimensioned Δ blocks make control synthesis and analysis cumbersome.

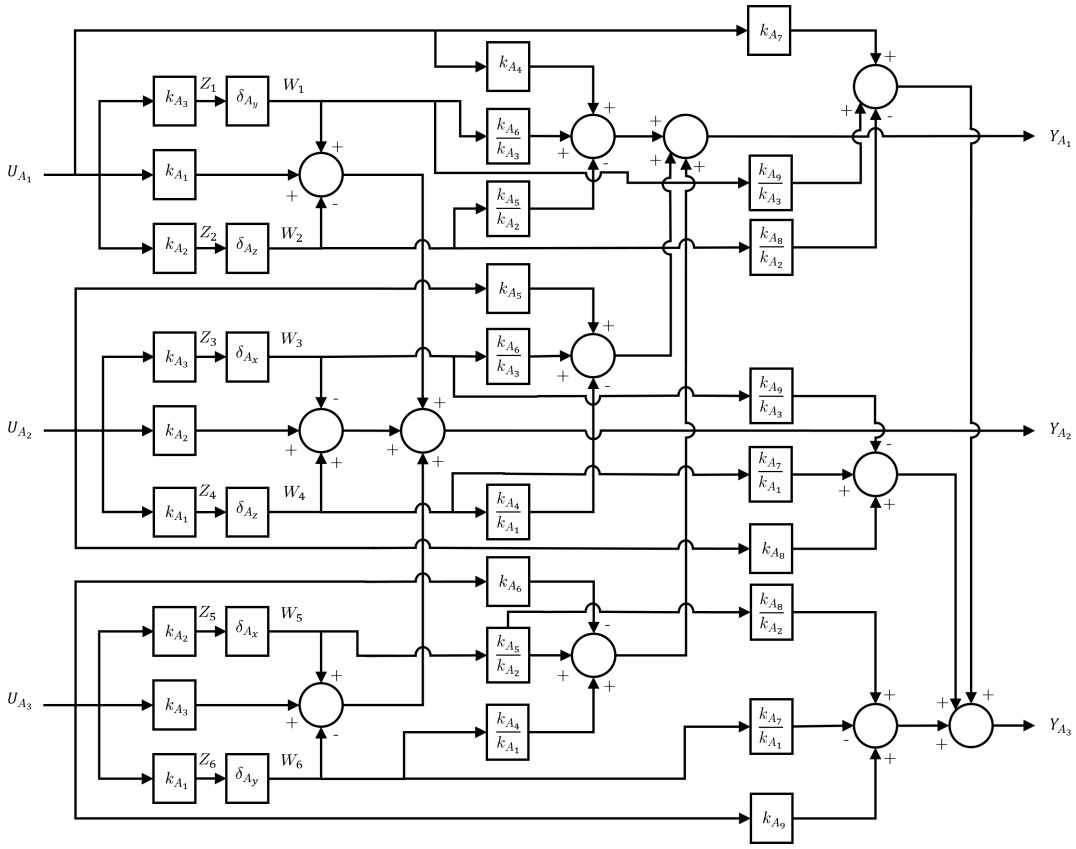
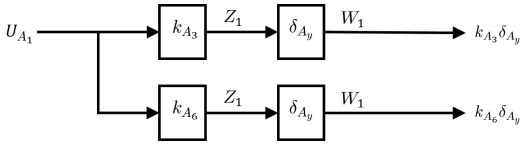
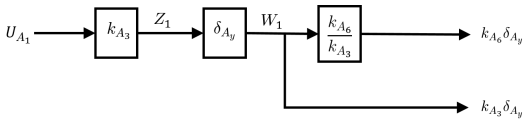


Figure 4.3: Block diagram representation of the linearised accelerometer system



(a) Redundant δ_{A_y} interconnection



(b) Minimal δ_{A_y} interconnection

Figure 4.4: Illustration of minimal interconnection of δ_{A_y} in the linearised accelerometer model

This type of manipulation is done throughout the block diagram interconnection, and the most minimal possible representation is arrived at. The inputs $W_1 - W_6$ are now assigned at the outputs of each occurrence of the uncertainties, with the outputs $Z_1 - Z_6$ at the inputs of the uncertainties i.e.,

$$\begin{aligned} W_1 &= \delta_{A_y} Z_1 \\ W_2 &= \delta_{A_z} Z_2 \\ W_3 &= \delta_{A_x} Z_3 \\ W_4 &= \delta_{A_z} Z_4 \\ W_5 &= \delta_{A_x} Z_5 \\ W_6 &= \delta_{A_y} Z_6 \end{aligned} \tag{4.6}$$

This step essentially isolates the uncertainties into the Δ block, while keeping M a certain matrix. The system is now formulated such that the outputs Y_A and Z are expressed in terms of the inputs U_A and W as $\begin{bmatrix} Z \\ Y_A \end{bmatrix} = M \begin{bmatrix} W \\ U_A \end{bmatrix}$, and is shown in

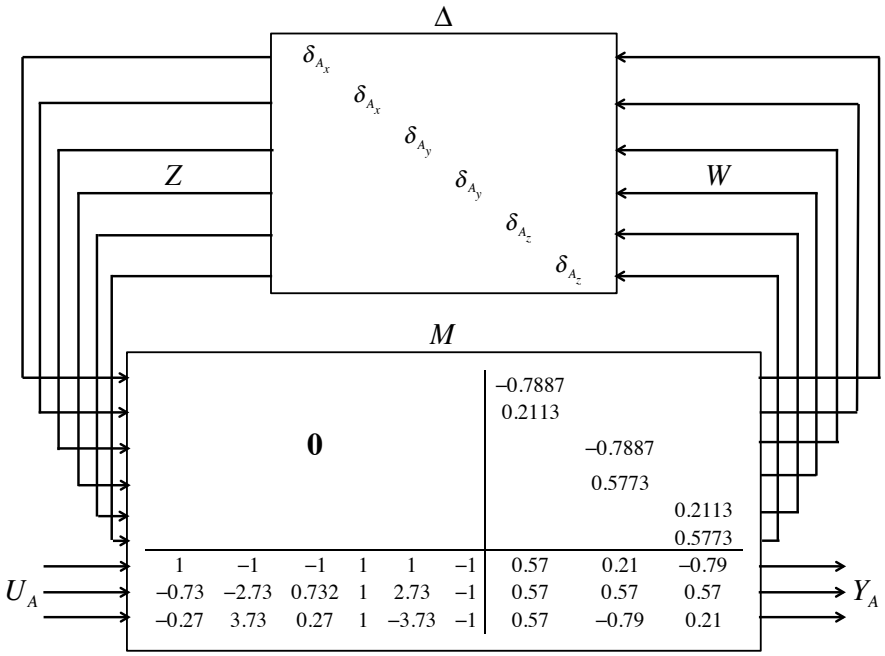


Figure 4.5: LFT formulation of the linearised accelerometer system

Equation (4.7). The linearised accelerometer LFT is shown in Figure 4.5.

$$\begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \\ \hline Y_{A_1} \\ Y_{A_2} \\ Y_{A_3} \end{bmatrix} = \left[\begin{array}{cccccc|ccc} & & & & & & k_{A_3} & 0 & 0 \\ & & & & & & k_{A_2} & 0 & 0 \\ & & & 0 & & & 0 & k_{A_3} & 0 \\ & & & & & & 0 & k_{A_1} & 0 \\ & & & & & & 0 & 0 & k_{A_2} \\ & & & & & & 0 & 0 & k_{A_1} \\ \hline 1 & -1 & -1 & 1 & 1 & -1 & k_{A_1} & k_{A_2} & k_{A_3} \\ \frac{k_{A_6}}{k_{A_3}} & -\frac{k_{A_5}}{k_{A_2}} & -\frac{k_{A_6}}{k_{A_3}} & \frac{k_{A_4}}{k_{A_1}} & \frac{k_{A_5}}{k_{A_2}} & -\frac{k_{A_4}}{k_{A_1}} & k_{A_4} & k_{A_5} & k_{A_6} \\ \frac{k_{A_9}}{k_{A_3}} & -\frac{k_{A_8}}{k_{A_2}} & -\frac{k_{A_9}}{k_{A_3}} & \frac{k_{A_7}}{k_{A_1}} & \frac{k_{A_8}}{k_{A_2}} & -\frac{k_{A_7}}{k_{A_1}} & k_{A_7} & k_{A_8} & k_{A_9} \end{array}\right] \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \\ W_5 \\ W_6 \\ \hline U_{A_1} \\ U_{A_2} \\ U_{A_3} \end{bmatrix} \tag{4.7}$$

In order for this LFT model to be implemented in Matlab, the uncertain parameters

must first be expressed. These are expressed via the RC toolbox's `ureal`[1]. Recall from Chapter 2 that uncertain parameters may be represented as, e.g., $\delta_{A_y} = \delta_{A_{y_0}} + r_{A_y} d_{A_y}$, with $\delta_{A_{y_0}}$ being the nominal value, r_{A_y} representing the range of variation of the parameter, and d_{A_y} being a random variable in [-1,1]. For a normalised parameter this may be written in Matlab as a `ureal` atom.

`ureal('delta_acc_x', 0,'Mode','PlusMinus','Range',[-1 1])` specifies a real uncertain parameter named `delta_acc_x` with a nominal value of 0 and a range of $[-1, 1]$[2]. The accelerometer lft is then included simply by specifying the M and Δ values and then performing an interconnection, which is shown as a code snippet in Figure 4.6. First, the partitioned blocks of M are specified, and are constituted into M. The Δ matrix is then specified in terms of the uncertain atoms. This completes the LFT $M - \Delta$ formulation for the accelerometer. It may be interconnected with other systems as appropriate.

The modelling technique described so far is a manual derivation from the block diagram interconnection of the model. A widely used and more straightforward technique is to simply allow Matlab's RC toolbox to carry out the uncertain atom interconnection, and then to use the `lftdata` command as in Figure 4.7. In this case the matrices R_A and D'_A are simply written down as they are and formulated as a state space model. The $M - \Delta$ matrices are then extracted using the function `lftdata`.

The latter method is clearly more straightforward, less error-prone and takes less effort to derive. The results of the two LFT methods must be compared. One way to do this for static gain systems is to subtract one system from another. If the subtraction yields a zero matrix, it is clear that the two systems are the same. Executing the system

[1] for real uncertain parameters, and `ucomplex` for complex uncertain parameters
[2] Recall that uncertain parameter values are normalized to $[-1, 1]$ in this study.

```
%Direction constants in D_acc
k1=0.5773; k2=0.2113;k3=-0.7887; k4=0.5773;
k5=0.5774; k6=0.5774; k7=0.5773; k8=-0.7887; k9=0.2113;

% Specifying the partitioned blocks of M
M12=[k3 0 0; k2 0 0; 0 k3 0; 0 k1 0; 0 0 k2; 0 0 k1];
M21=   [1 -1 -1 1 1 -1;
        k6/k3 -k5/k2 -k6/k3 k4/k1 k5/k2 -k4/k1;
        k9/k3 -k8/k2 -k9/k3 k7/k1 k8/k2 -k7/k1];
M22=[k1 k2 k3; k4 k5 k6; k7 k8 k9];

%Express M and Delta matrices
ACC_SEN.M=[zeros(6) M12; M21 M22];
ACC_SEN.Delta = [delta_acc_y zeros(1,5);
                         0 delta_acc_z zeros(1,4);
                         0 0 delta_acc_x zeros(1,3);
                         0 0 0 delta_acc_z zeros(1,2);
                         0 0 0 0 delta_acc_x 0;
                         0 0 0 0 0 delta_acc_y];
```

Figure 4.6: Code snippet with manually derived $M - \Delta$ formulation of the linearised accelerometer model.

```
%Specify D_acc and R_acc
D_acc = [0.5773    0.2113   -0.7887;
         0.5773    0.5774    0.5774;
         0.5773   -0.7887    0.2113];

R_acc = [1 angle_acc_z -angle_acc_y;
        -angle_acc_z 1 angle_acc_x;
         angle_acc_y -angle_acc_x 1];

%Multiply to get the output gain matrix
ACC_D = D_acc*R_acc;

%Get the state space representation and express as LFT
ACC_SEN.name='Actuator model';
ACC_SEN.model = ss(ACC_D);
[ACC_SEN.M2,ACC_SEN.Delta2]=lftdata(ACC_SEN.model);
```

Figure 4.7: Code snippet showing $M - \Delta$ extraction of the linearised accelerometer model using the RC Toolbox.

```
compare = lft(ACC_SEN.Delta,ACC_SEN.M)-ACC_SEN.model  ;
compare

ans =

  d =
       u1  u2  u3
   y1   0   0   0
   y2   0   0   0
   y3   0   0   0

Static gain.
```

Figure 4.8: Comparison of the manually derived and automatically generated LFT formulations

subtraction in Matlab's command window, the result is as shown in Figure 4.8.

Thus, the static gain of zero with no uncertainties, shows that the two systems are identical. Matlab's modelling, therefore, gives the same result with far less effort and without cumbersome block diagram manipulations. For the rest of the study therefore, a direct Matlab interconnection is done for LFTs, except when specified. Each model is first created in Matlab, its $M - \Delta$ formulation extracted via `lftdata`, and validated. The validation strategy is now described.

4.3.3 LFT model validation

It is vital for both controller synthesis and analysis to ensure that the generated LFT model has a response close to that of the original uncertain system. For dynamical models, a frequency domain approach was employed to validate the low level LFTs. For each LFT model, 20 randomly perturbed LTI systems within the uncertainty bounds are first obtained. The singular values of these systems were computed and plotted with the uncertain samples of the original LTI models. The MATLAB validation uses the command `usample` for generating random samples of an uncertain system. The code snippet for validating LFTs is shown in Figure 4.9.

```
samp_system=usample(system.model,20); %Generate 20 samples of original uss

lft_system=lft(system.Delta,system.M); %Interconnect M-Delta as upper LFT

samp_lft_system=usample(lft_system,20); %Generate 20 samples of LFT

sigma(samp_system,samp_lft_system,{0.0001,100000}); %Singular value plot
```

Figure 4.9: Validating dynamical system LFT formulations in Matlab.

Most sensors and actuators are not dynamical systems — instead they are static gain systems. These are validated similarly to the accelerometer described above in Figure 4.8. Comparison is done by subtracting the original uncertain system and the generated LFT model. A zero output gain matrix indicates identical systems, while non-zero systems must be rechecked for modelling errors.

Having discussed how to construct LFT formulations of models, the next sections proceed to describe the spacecraft dynamics and their LFT modelling.

4.3.4 Integrated Spacecraft Dynamics

The chaser spacecraft dynamical models are presented in this section. These dynamics are standard for a flexible spacecraft with effects of fuel sloshing included, and can be found in standard texts on the subject, e.g, [111]. A few simplifying assumptions are made, which will be mentioned. The LFT modelling of the resulting system is also presented here. One of the key challenges in spacecraft control law development is the modelling of flexible appendages, whose effects are difficult to model accurately. The chaser spacecraft has large solar arrays to supply energy for the long flight from Earth to Mars. These solar arrays cause forces and torques to act on the spacecraft rigid body. These are not known accurately, but the dynamics are physically modelled in the form of flexible modes. Their frequency and damping and difficult to model accurately, and

so are specified as uncertain parameters. The dynamics for each flexible mode is:

$$B_{T,n_F}^T \dot{v}_{sc} + B_{R,n_F}^T \dot{\Omega}_{sc} + m_{n_F} \ddot{q}_{n_F} + l_{n_F} \dot{q}_{n_F} + k_{n_F} q_{n_F} = 0 \tag{4.8}$$

where v_{sc} is the linear velocity of the SC centre of mass, Ω_{sc} is the SC angular velocity. q_{n_F} is the coordinate of each mode, $l_{n_F} = 2m_{n_F}\zeta_{n_F}\omega_{n_F}$ is the dissipative coefficient, with ζ_{n_F} being the mode's damping ratio, and ω_{n_F} is the mode's natural frequency. $k_{n_F} = m_{n_F}\omega_{n_F}^2$ is the stiffness coefficient. B_{T,n_F} is the participation coefficient of each mode in the spacecraft translational motion. B_{R,n_F} is the modal participation in rotation at the spacecraft centre of mass for mode n_F. These parameter values are estimated using structural analysis via finite element methods [111].

The chaser SC has large propellant tanks to fuel the long journey to Mars. When these tanks are partially full, the effects of fuel slosh are significant. These effects are difficult to model accurately, and is an active research area [113]. In the current study, the effects of slosh may be modelled in the form of a linear pendulum, although different approaches exist such as the spring-mass model of sloshing [114].

Each propellant tank is assumed to consist of a fixed mass of fuel that does not move during thrust, and a linear pendulum representing the motion of the propellant mass that participates in the the slosh motion during thrust. This is shown in Figure below. Assuming a total of n_S modes over all the fuel tanks, each mode dynamics is given by:

$$B_{T,n_S}^T \dot{v}_{sc} + B_{R,n_S}^T \dot{\Omega}_{sc} + m_{n_S} \ddot{\theta}_{n_S} + l_{n_S} \dot{\theta}_{n_S} + k_{n_S} \theta_{n_S} = 0, \tag{4.9}$$

where θ_{n_S} is the sloshing mode coordinate. Note that this is similar to the solar ar-

ray dynamics (4.8), since both are considered as flexible appendages that impact the rigid body motion. B_{T,n_S} is the participation coefficient of each sloshing mode in the spacecraft translational motion. B_{R,n_S} is the modal participation in rotation at the spacecraft centre of mass for sloshing mode n_S. θ_{n_S} is the coordinate of each mode, $l_{n_S} = 2m_{n_S}\zeta_{n_S}\omega_{n_S}$ is the dissipative coefficient, with ζ_{n_S} being the mode's damping ratio, and ω_{n_S} is the mode's natural frequency. $k_{n_S} = m_{n_S}\omega_{n_S}^2$ is the stiffness coefficient. These parameters are derived using the principles of fluid mechanics via finite element analysis.

Note that the structure of both (4.8) and (4.9) are similar to that of a second order spring mass damper model — also known as the MLK model[1]. Including these in the spacecraft translation and rotational motion, we obtain:

$$F_{sc} = m_{sc}\dot{v}_G + \sum_{n_F} \ddot{q}_{n_F} B_{T,n_F} + \sum_{n_S} B_{T,n_S}\ddot{\theta}_{n_S} \tag{4.10}$$

$$T_{sc} = J_{sc}\dot{\Omega} + \sum_{n_F} \ddot{q}_{n_F} B_{R,n_F} + \sum_{n_S} \ddot{\theta}_{n_S} B_{R,n_S} \tag{4.11}$$

where F_{sc} and T_{sc} are respectively the forces and torques acting on the spacecraft. These equations may be combined and rearranged into a composite MLK model form as follows:

$$M_{sc}\ddot{X} + L_{sc}\dot{X} + K_{sc}X = \underline{\mathrm{B}}_{sc}U, \tag{4.12}$$

where the states are $X = \begin{bmatrix} x & \alpha & q & \theta \end{bmatrix}^T$, the inputs are $U = \begin{bmatrix} F_{sc} & T_{sc} \end{bmatrix}^T$ and the

[1] with M represnting mass, L representing damping coefficient and K representing spring stiffness constant

input distribution matrix is $\underline{\mathrm{B}}_{sc} = \begin{bmatrix} I_3 & 0 \\ 0 & I_3 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$.

x and α are the displacement and angular displacement of the spacecraft respectively, while q represents the flexible mode co-ordinate vector over n_F modes, and θ represents the sloshing coordinates vector over n_S modes. The matrices M_{sc}, L_{sc} and K_{sc} are analogous to the generalised mass, damping factor and stiffness coefficient taken over the entire spacecraft rigid body with the effects of solar array flexible modes and sloshing modes included. These matrices are derived in the literature [110]. The generalised mass matrix is written as:

$$\underline{\mathrm{M}}_{sc} = \begin{bmatrix} m_{sc}I_3 & 0 & \underline{\mathrm{B}}_{T,F} & \underline{\mathrm{B}}_{T,S} \\ 0 & J_{sc} & \underline{\mathrm{B}}_{R,F} & \underline{\mathrm{B}}_{R,S} \\ \underline{\mathrm{B}}_{T,F}^T & \underline{\mathrm{B}}_{R,F}^T & \underline{\Lambda}_F & 0 \\ \underline{\mathrm{B}}_{T,S}^T & \underline{\mathrm{B}}_{R,S}^T & 0 & \underline{\Lambda}_S \end{bmatrix}, \tag{4.13}$$

where $\underline{\mathrm{B}}_{T,F}$ and $\underline{\mathrm{B}}_{T,S}$ are the translational participation matrices due to solar arrays and sloshing respectively, while $\underline{\mathrm{B}}_{R,F}$ and $\underline{\mathrm{B}}_{R,S}$ are the respective rotational participation matrices. $\underline{\Lambda}_F$ and $\underline{\Lambda}_S$ are called the effective mass matrices due to flexible modes and sloshing respectively. The generalised damping factors including the effects of rigid body and appendages is:

$$\underline{\mathrm{L}}_{sc} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & l_F & 0 \\ 0 & 0 & 0 & l_S \end{bmatrix}. \tag{4.14}$$

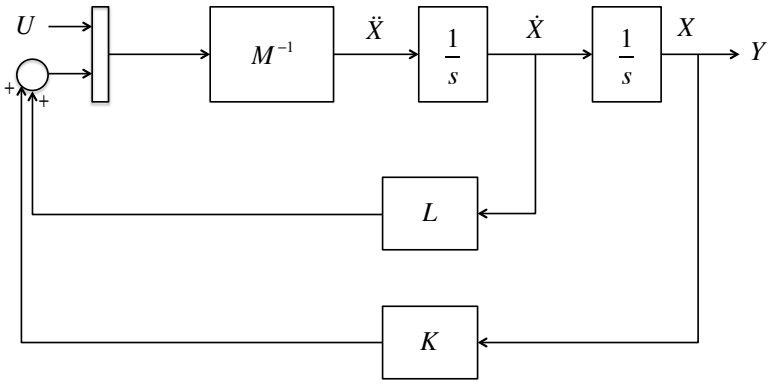


Figure 4.10: Spring mass damper block diagram interconnection

where l_F includes the damping factors from over all the solar array appendage modes, and l_S includes damping factors from over all the sloshing modes in all fuel tanks. Similarly, the generalised stiffness coefficient is

$$\underline{\mathrm{K}}_{sc} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & k_F & 0 \\ 0 & 0 & 0 & k_S \end{bmatrix}. \tag{4.15}$$

with k_F representing the spring stiffness coefficient from the solar array modes, and k_S representing spring stiffness from over all the sloshing modes in all fuel tanks. The interconnection of the MLK system is the familiar second order state space system, and is represented in Figure 4.17 below.

Several of the quantities in the integrated spacecraft model are unknown or are incompletely modelled. These are represented as uncertain parameters. In order for robust control design and analysis to be carried out, the uncertain system must be presented in the form of LFT. This is done using the physical modelling technique, by specifying the MLK block diagram equations, and performing their interconnection. Matlab's robust

```
% Specify block names, inputs and outputs
systemnames = 'invM K_sc L_sc inp_1 inp_2 BB';
inputvar = '[F_sc{3}; T_sc{3}]';
outputvar = '[ inp_2(4:6) ]';

% Specify inputs to each block
input_to_BB = '[F_sc; T_sc]';
input_to_K0 = '[inp_2]';
input_to_L0 = '[inp_1]';
input_to_int_1 = '[invM]';
input_to_int_2 = '[inp_1]';
input_to_invM = '[BB - K_sc - L_sc]';

% Clean up sysic and perform interconnection
sysoutname = 'sc_dyn';
cleanupsysic = 'yes';
sysic;
```

Figure 4.11: Spring mass damper block diagram interconnection

control toolbox performs uncertain system interconnection using the `sysic` function. A code snippet showing the system interconnection of the MLK system is shown in Figure 4.11.

The resulting system is an uncertain state-space system with 3 inputs and 6 outputs with 184 occurrences of uncertainties. The number of unique uncertainties is 39. The model can be expressed in the $M - \Delta$ form using Matlab's `lftdata` command, simply as `[Msc,Deltasc] = lftdata(scdyn)`. The resulting Deltasc matrix is a 184×184 diagonal matrix, and Msc is a 187×190 matrix. The Mass, COG and Inertia (MCI) elements are not perfectly known due to construction and uncertainty in the mounting distances and masses of the different elements of the spacecraft. The uncertain parameters, their description and the number of their occurrences is shown in Table 4.1.

The LFT is validated by plotting the singular values of the LFT and the original system, as described in the previous section. This plot is shown in Figure below. It may be observed that there is a very close matching of the responses based on the 20 random uncertain LTIs generated. The frequency range considered is the frequency

Table 4.1: Uncertain parameters in the spacecraft dynamics LFT model

Type	Description	Name	Occurrences	Nominal Value	Range
MCI	Mass of spacecraft [kg]	*dmass_sc*	3	2500	230
	Centre of gravity [m]	dCoGx	4	1.55	0.03
		dCoGy	4	0.005	0.03
		dCoGz	4	-0.03	0.03
	Inertia [kg/m/m]	dIxx	1	3810	762
		dIyy	1	2500	500
		dIzz	1	5540	1108
Solar Arrays	Solar array angle [rad]	*dangle_sa*	16	0	2.5
	Damping [-]	*ddamp_sa_*1	2	0.003	0.002
		*ddamp_sa_*2	2	0.003	0.002
		*ddamp_sa_*3	2	0.003	0.002
		*ddamp_sa_*4	2	0.003	0.002
	SA frequency [rad/s]	*domega_sa_*1	6	0.903	0.1806
		*domega_sa_*2	6	4.419	0.8838
		*domega_sa_*3	6	6.767	1.353
		*domega_sa_*4	6	6.92	1.384
	SA rotation participation[-]	$dBR_sa\_1\_x$	4	36.64	0
		$dBR\_sa\_2\_y$	2	0.06	0.003
		$dBR\_sa\_2\_z$	4	-37.01	-1.8505
		$dBR\_sa\_3\_x$	4	3.16	0.158
		$dBR\_sa\_3\_y$	2	0.23	0.0015
		$dBR\_sa\_3\_z$	4	0.03	0.0015
		$dBR\_sa\_4\_x$	4	-0.19	-0.0095
		$dBR\_sa\_4\_y$	2	3.74	0.187
		$dBR\_sa\_4\_z$	4	0.02	0.001
Sloshing	Sloshing Mass [kg]	*dmass_sl*1	8	180	2
		*dmass_sl*2	8	280	3
	Damping [-]	*ddamp_sl*1	2	0.01	0.0005
		*ddamp_sl_*2	2	0.01	0.0005
	Pendulum length	*dlength_sl*1	16	0.29	0.0029
		*dlength_sl*2	16	0.29	0.0029
	Slosh frequency [rad/s]	*domega_sl*1	6	0.8384	0.04192
		*domega_sl*1	6	0.8384	0.04192
	SA translation participation [-]	$dBT\_sa\_1\_y$	4	-0.01	-0.0005
		$dBT\_sa\_1\_z$	2	6.54	0.327
		$dBT\_sa\_2\_x$	4	6.79	0.3395
		$dBT\_sa\_3\_y$	4	-0.04	-0.002
		$dBT\_sa\_4\_z$	2	3.11	0.1555
		$dBT\_sa\_1\_z$	4	-0.18	-0.009

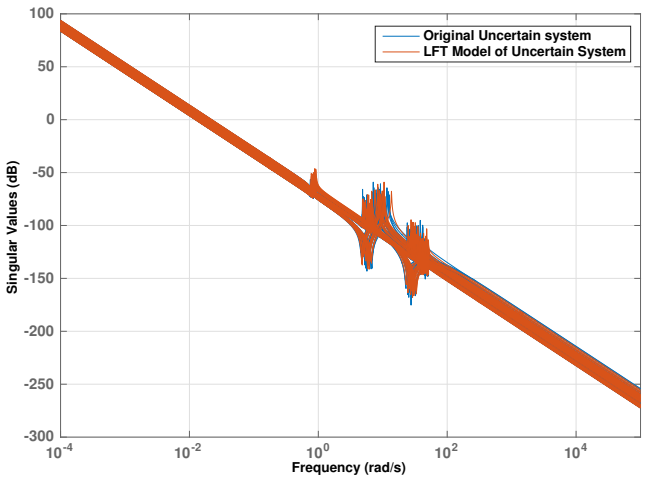


Figure 4.12: Singular value comparison between integrated spacecraft dynamics and its LFT model

range considered in the control design. Observe the peaks in the middle frequency range, between about 1 rad/s and 100 rad/s. This is expected to be caused due to the flexible modes and sloshing effects.

Revisiting the LFT model, the mass matrix M_{SC} is the component having the most impact on the size of the system.in terms of the number of instances of uncertain parameters. m_G, J_G, μ_{FA}, μ_{SL} and the participation factors $B_{T,FA}$, $B_{R,FA}$, $B^T_{T,SL}$ and $B_{R,SL}$ are the uncertain parameters. The flexible appendages uncertain parameters for B_T and B_R are specified in terms of the appendage frame. For all computations, they must be expressed in the spacecraft body frame. They need to be transformed into the body frame, which involves (i) rotation of the appendage frame with respect to the main body frame and (ii) translating the origin to the rigid body centre of gravity. For

example, for the flexible modes:

$$B_{T,n_F} = \mathcal{R}_{n_F} \bullet B_{T,A,n_F} \tag{4.16}$$

$$B_{R,n_F} = GA \times B_{T,A,n_F} + \mathcal{R}_{n_F} \bullet B_{R,A,n_F} \tag{4.17}$$

The rotation from the appendage frame to the chaser body frame carried out via the rotation matrix. The rotation matrix is written as

$$\mathcal{R}_{n_F} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\psi & -sin\psi \\ 0 & sin\psi & cos\psi \end{bmatrix} \tag{4.18}$$

where ψ is the uncertain appendage angle. ψ can vary in $[0^0, 360^0]$ and hence the rotation matrix model cannot be linearised by a Taylor's series expansion with a small-angle assumption. In order to accommodate for this large variation in ψ, a transformation is done as described in [112]. The sine and cosine terms may be expressed in terms of $tan\psi$:

$$\begin{aligned} sin\psi &= \frac{2tan\frac{\psi}{2}}{1 + tan^2\frac{\psi}{2}} \\ cos\psi &= \frac{1 - tan^2\frac{\psi}{2}}{1 + tan^2\frac{\psi}{2}} \end{aligned} \tag{4.19}$$

The uncertainty in ψ is reported in the $tan\frac{\psi}{2}$ term as follows:

$$\tan\frac{\psi}{2} = \tan\frac{\psi_0}{2} + \delta_\psi \tag{4.20}$$

where ψ_0 is 0^0 and $\delta_\psi \in [0, +\infty[$. The sine and cosine terms are then transformed to:

$$sin\psi = \frac{2\delta_\psi}{1+\delta_\psi^2} \qquad cos\psi = \frac{1-\delta_\psi^2}{1+\delta_\psi^2} \tag{4.21}$$

The uncertainties are required to be normalized such that they vary in $]-1, 1[$. This is done via two transformations. Firstly, $\delta_\psi = \frac{\delta'_\psi}{1-\delta'_\psi}$ so that δ'_ψ varies in the range $[0, 1]$ and next, $\delta_\Re = 2\delta'_\psi - 1$, where $\delta_\Re$ varies in the $]-1, 1[$ range.

These transformations upon simplification yield:

$$\sin\psi = \frac{\delta_\Re^2}{1+\delta_\Re^2} \qquad \cos\psi = \frac{-2\delta_\Re}{1+\delta_\Re^2} \tag{4.22}$$

The rotation matrix in Equation (4.3.4) is therefore written as:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{-2\delta_\Re}{1+\delta^2{}_\Re} & -\frac{1-\delta^2{}_\Re}{1+\delta_\Re^2} \\ 0 & \frac{1-\delta^2{}_\Re}{1+\delta_\Re^2} & \frac{-2\delta_\Re}{1+\delta^2{}_\Re} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} \tag{4.23}$$

where U_1, U_2, U_3 are the inputs and Y_1, Y_2, Y_3 are the outputs. In order to formulate this in the $M - \Delta$ form, it is necessary to draw the block diagram interconnection as before, and choose the internal inputs W and internal outputs Z such that $W = \Delta Z$. Again, these must be chosen carefully - a poor choice of the internal inputs and outputs can lead to a large LFT with several occurrences of the uncertainty δ_2. For large complicated models such as the one under consideration, it is important to create LFTs that are as minimal as possible. This is because, upon interconnection of the LFTs, these uncertainties can add up to a large number, making the model computationally intensive. Upon examining the block diagram for this system, four internal inputs

W_1, W_2, W_3, W_4 and four internal outputs Z_1, Z_2, Z_3, Z_4 are chosen. The resulting LFT is shown in Equation (4.26).

$$\begin{bmatrix} Z \\ Y \end{bmatrix} = \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right] \begin{bmatrix} W \\ U \end{bmatrix} \tag{4.24}$$

and

$$W = \Delta Z \tag{4.25}$$

i.e.

$$\left[\begin{array}{c} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ \hline Y_1 \\ Y_2 \\ Y_3 \end{array}\right] = \left[\begin{array}{cccc|ccc} 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -2 & 0 & 0 & 2 & 0 & 0 & -1 \\ 0 & -2 & -2 & 0 & 0 & 1 & 0 \end{array}\right] \left[\begin{array}{c} W_1 \\ W_2 \\ W_3 \\ W_4 \\ \hline U_1 \\ U_2 \\ U_3 \end{array}\right] \tag{4.26}$$

and

$$\begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} \delta_2 & 0 & 0 & 0 \\ 0 & \delta_2 & 0 & 0 \\ 0 & 0 & \delta_2 & 0 \\ 0 & 0 & 0 & \delta_2 \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \end{bmatrix} \tag{4.27}$$

Equation (4.27) represents the required minimal form of the LFT model. This represen-

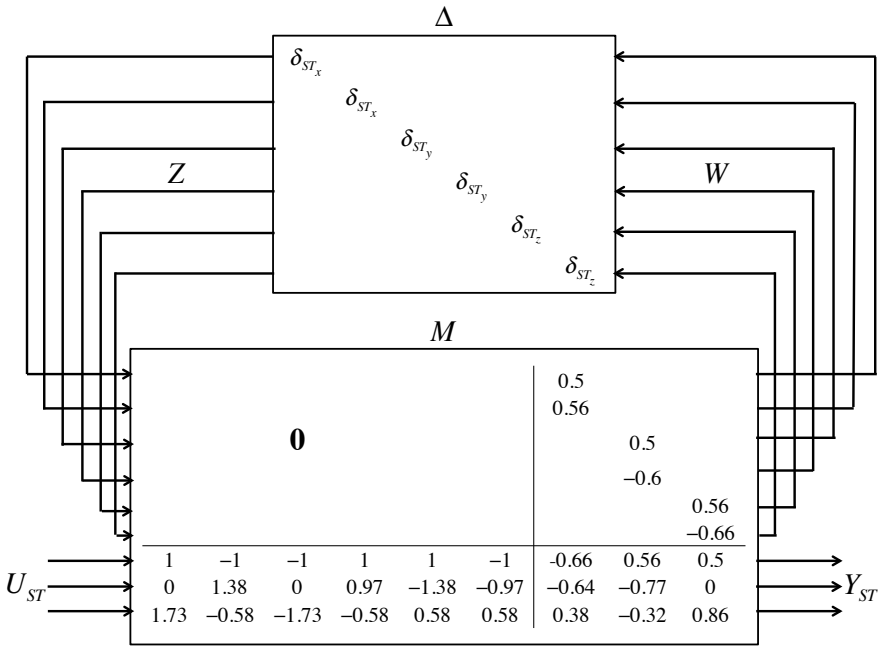


Figure 4.13: Star tracker LFT interconnection

tation is now included in the spacecraft dynamics. Similarly, LFT models are created for several other systems. We report here one example each of sensors and actuators.

4.3.5 Sensor modelling

The sensor system chosen here is the star tracker. The star tracker is an optical sensor responsible for measuring the attitude of the SC with high accuracy around all three spacecraft axes, which is crucial for rendezvous.The star tracker assembly consists of two star trackers for accuracy as well as redundancy purposes.

This model is similar in structure to the accelerometer. Essentially, the measurement provided by the sensor must undergo a transformation to account for the misalignment angles.

4.3.6 Inclusion of Time Delays

The effects of time delays are important, and need to be considered in the analysis, particularly in sensors and actuators. A time delay not properly considered has a risk of computing incorrect control inputs. In this system, the sensor and actuator time delays are not known with precision and hence are to be considered as uncertain parameters. In the iGNC framework, they are included in the star tracker, Narrow-Angle Camera (NAC) and for the thrusters.

The Padé first order approximation [15] is made use of for incorporating time delays. A time delay of τ_d seconds enters the system as an exponential term $e^{-\tau_d s}$. The Padé approximation of the exponential rate term gives:

$$e^{-\tau_d s} = \frac{1 - \frac{\tau_d s}{2}}{1 + \frac{\tau_d s}{2}} = \frac{2 - \tau_d s}{2 + \tau_d s} \tag{4.28}$$

This time-delay system is represented in the block diagram in Figure 4.14.

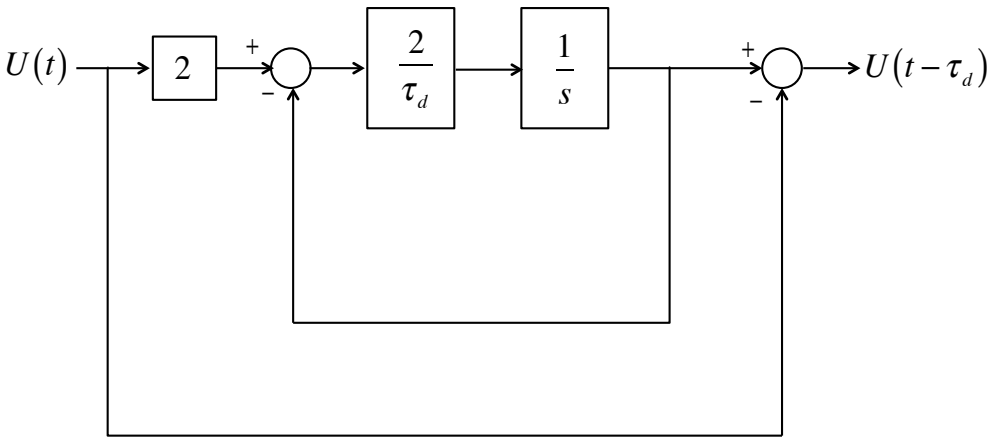


Figure 4.14: Block diagram representation of the approximated time-delay system.

Them, the time delay dynamics may be represented by the following state equations:

$$\dot{X} = \frac{-2}{\tau_d} X + \frac{4}{\tau_d} U \tag{4.29}$$
$$Y = X - U$$

This may be written as the plant $G_d(s)$:

$$G_d(s) = \left[\begin{array}{c|c} \frac{-2}{\tau_d} I & \frac{4}{\tau_d} I \\ \hline I & -I \end{array}\right] \tag{4.30}$$

where the size of the identity matrices I is given by the number of inputs to the plant. The uncertain time delay is represented as follows:

$$\tau_d = \bar{\tau} + r_\tau \delta_\tau \tag{4.31}$$

Notice that there is a singularity in τ_d in the delay system. This means that the time delay should necessarily only be modelled for cases that require it, since a zero or a very low delay value could cause ill-conditioning in the model in 4.30. The $M - \Delta$ representations of each of the matrices is derived and presented in the block diagram in Figure 4.15.

where

$$M_{B_\tau} = \left[\begin{array}{cc} \frac{-r_\tau}{\tau} I & \frac{r_\tau}{\tau} I \\ \frac{-4}{\tau} I & \frac{4}{\tau} I \end{array}\right], \tag{4.32}$$

$$M_{A_\tau} = \left[\begin{array}{cc} \frac{-r_\tau}{\tau} I & \frac{r_\tau}{\tau} I \\ \frac{2}{\tau} I & \frac{-2}{\tau} I \end{array}\right], \tag{4.33}$$

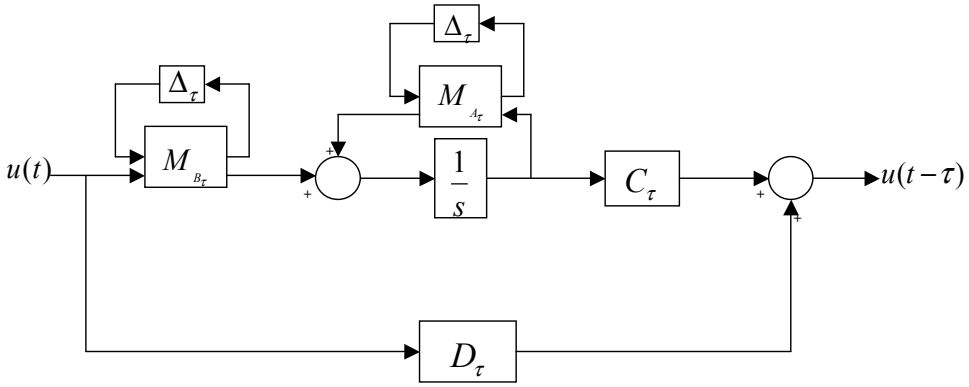


Figure 4.15: Block diagram representation of the state space matrices with uncertain time delay included.

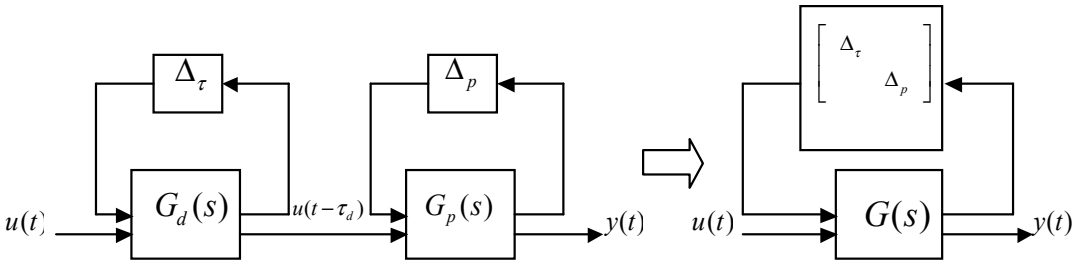


Figure 4.16: Series LFT connection of the approximated delay system to an LFT

and $\Delta_\tau = \delta_\tau I$. The delay model is included in series with the plant model (say $G_p(s)$ with uncertain parameter block Δ_p), which is a standard LFT operation, and also leads to an LFT as in Figure 4.16.

This delay model is then connected to the sensor model, in this case the star tracker model. The star tracker delay is in the range [100,300]ms, i.e., a nominal value of 200ms with a variability of 100ms. Upon interconnecting the delay LFT model, and substituting these values, the step response is as follows. A small peak of the order 10^{-17} is observed in all three input-output channels, which settles in about 0.6 seconds.

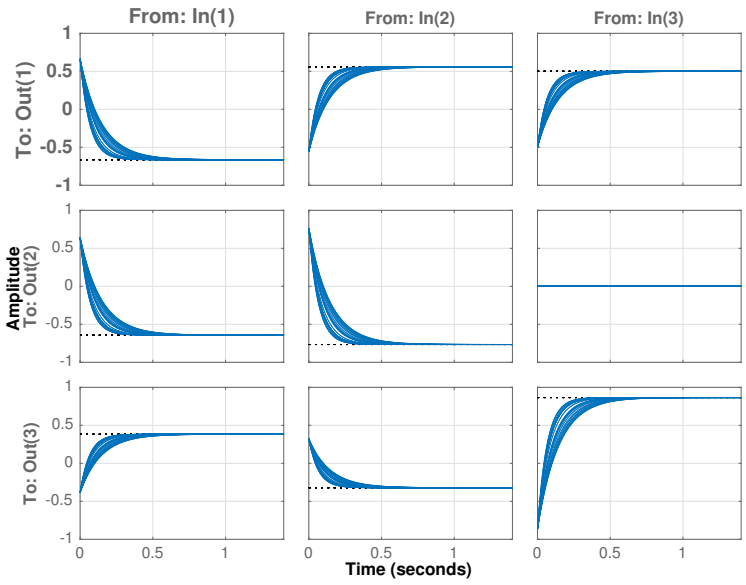


Figure 4.17: Star tracker step response with time delay included

This approximation is acceptable for the purposes of designing a controller.

4.3.7 Actuator modelling

In this section, the reaction wheel LFT modelling is described. The reaction wheel is used as the actuator for small angle attitude control. The reaction wheel assembly provides the SC angular momentum due to reaction wheel load , expressed in the chaser body frame.

As in the previous case, the true response of the star tracker is modelled by multiplying the observed attitude values by a transformation matrix accounting for the misalignment angles on the mounting frame. This is modelled as follows:

$$D_{RW} = R_{RW} * D'_{RW} \tag{4.34}$$

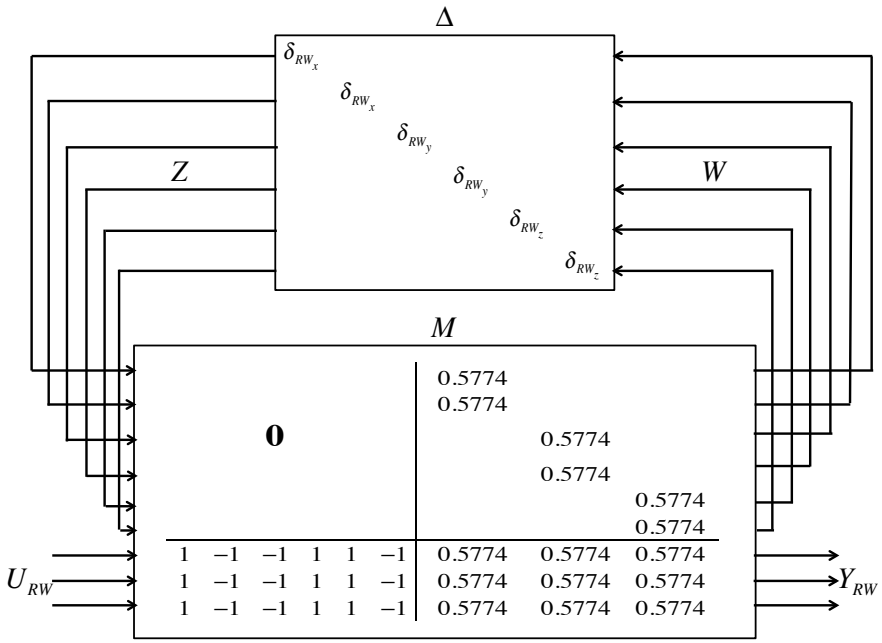


Figure 4.18: Reaction wheel LFT interconnection

with D'_{RW} being a 3×3 matrix with constant values representing the directions. R_{RW} represents the uncertain misalignment on the mounting frame, similar to the accelerometer case. This gives an LFT $M-\Delta$ model of the same structure as the last two systems, but with different values in the M matrix. This is shown in Figure 4.3.7.

.

4.4 Linear analysis environment

The above LFT models are useful in control design, and in analysing the effects of specific subsystems independent of others. The full simplified model for analysis, however, is created by the consortia and sent to Exeter. The general idea is to create a block diagram interconnection in Simulink, and then to linearise it using Matlab's tools. This is considered to be the design stage analysis, and so only chaser subsystems are considered. The Simulink interconnection is shown below in Figure 4.19.

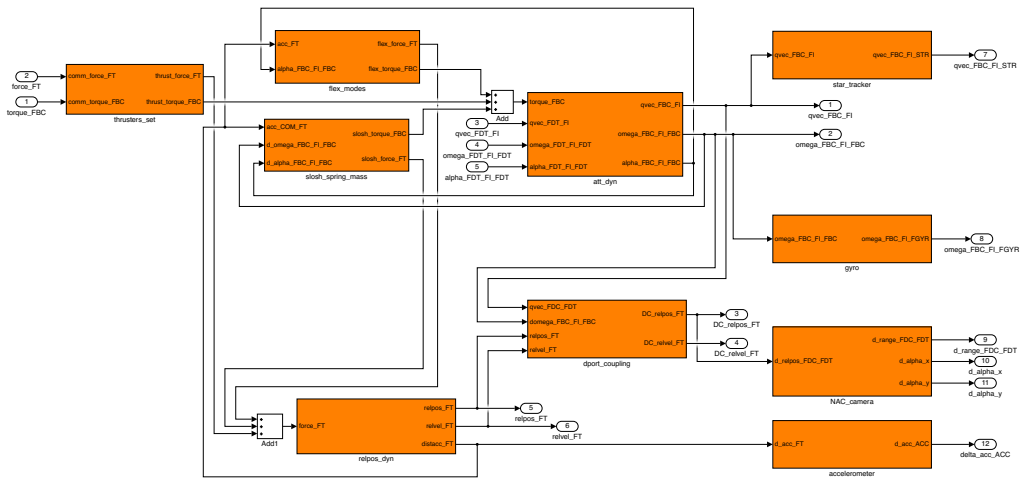


Figure 4.19: Simplified rendezvous model block diagram interconnection in Simulink.

```
TRDC_openloop = ulinearize('iGNC_mode_TRDC_plant');
[TRDC_M,TRDC_delta] = lftdata(TRDC_openloop);
```

Figure 4.20: Creating linearised models and extracting their LFT formulations in Matlab

The linearisation of this plant is then carried out in Matlab. The RC toolbox function `ulinearize` linearises a Simulink model consisting of uncertain state space blocks. This linearisation is done without evaluating the uncertain variables, and so the process is not affected by uncertainty values or structure. Once the linearised plant is extracted, its LFT is computed as before, using the `lftdata` function as in Figure 4.20. The result is an uncertain state-space system, with 30 inputs and 15 outputs. The uncertainties present, its system response and detailed performance analysis are presented in the next chapter.

The analysis in this linear design environment is detailed in Chapter 5. This analysis is carried out using a combination of classical methods (such as analytical methods and optimisation methods), as well as the newly developed probabilistic methods. In the following sections, a performance analysis of the full, high-fidelity nonlinear model of the rendezvous model is carried out. First, a brief description of this model is given. Note that this is a black-box system in the true sense, since most of the files are protected by GMV and cannot be accessed. The access is largely limited to the uncertain parameter values and output signal values.

4.5 iGNC Real World Model

The high-fidelity real world model used for analysis is described in this section. The initialisation, set up and analysis is done via GMV's proprietary software called Guidance, Navigation and Control Development Environment (GNCDE) [115]. GNCDE is a suite for controller synthesis and analysis, which includes a large library of models of

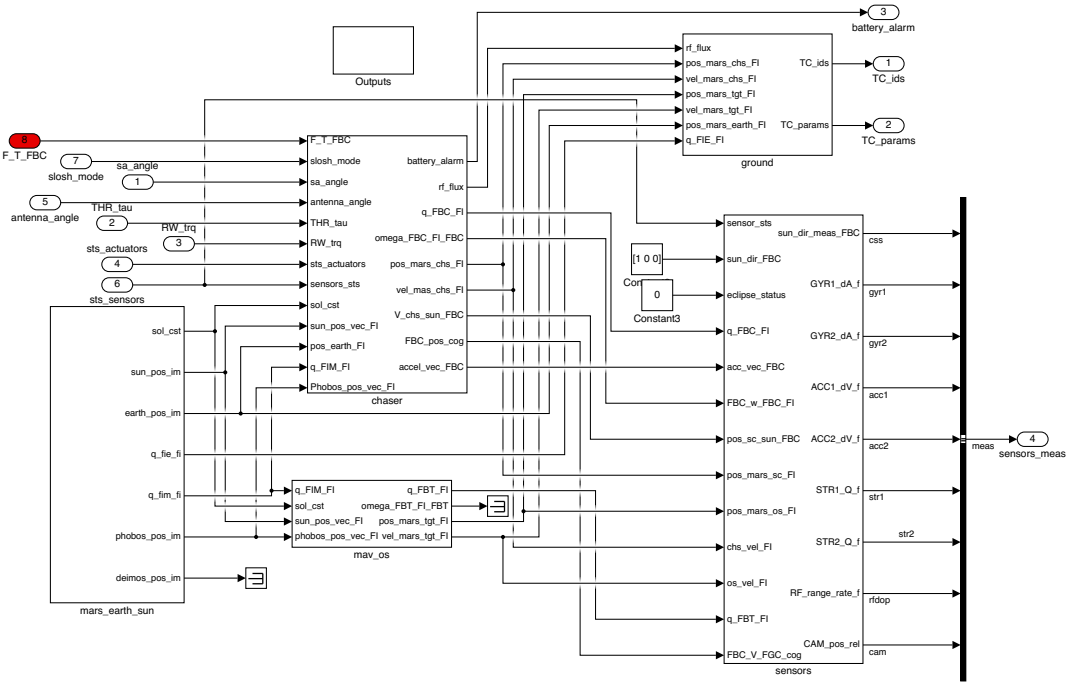


Figure 4.21: Block diagram representation of the high-fidelity rendezvous model

spacecraft dynamics, sensors, planetary mechanics and telecommunication.

Specific missions are included in the form of templates [115], whose subsystems, uncertain values or parameters can be modified. The mission template includes nonlinear models, and is devoted to high fidelity performance assessment activities. The iGNC mission high-level view of the models is shown below in Figure 4.21. Observe that the interconnection consists of all relevant models, including the MAV and OS separation, orbital mechanics of Mars, Earth and Sun[1], all sensors and ground communication, apart from the chaser dynamics.

As a result, a large number of uncertainties are considered in the analysis. Notice

[1] including effects of all their gravity, which is a significant factor in interplanetary transfer

that several of the uncertainties are due to thruster misalignment, owing to 6 thruster assemblies of 4 thrusters each. Every thruster is affected by the misalignment in $x-$, $y-$ and $z-$ axes on its mounting frame. The uncertainties considered are shown in the Table 4.2 below.

Table 4.2: Uncertain parameters in the high-fidelity iGNC simulation model

Uncertainty	Description	Unit	Nominal Value	Variation	Range
dinit_posx	Initial relative position	m	-100	5	[-105; -95]
dinit_posy		m	0	5	[-5; 5]
dinit_posz		m	0.113	5	[-5.1645; 4.8355]
dinit_velx	Initial relative velocity	m/s	0.1	0.05	[0.05; 0.15]
dinit_vely		m/s	0	0.15	[-0.15; 0.15]
dinit_velz		m/s	0	0.05	[-0.05; 0.05]
dtotal_mass	Mass of the SC	kg	2300	230	[2070; 2530]
dIxx	SC Inertia	kg m2	420	20%	[336; 504]
dIxy		kg m2	-10	20%	[-8; -12]
dIxz		kg m2	230	20%	[184; 276]
dIyy		kg m2	2330	20%	[1864; 2796]
dIyz		kg m2	10	20%	[8; 12]
dIzz		kg m2	2040	20%	[1632; 2448]
dflex_Ixx	Solar panel inertia	kg m2	291	20%	[232.8; 349.2]
dflex_Iyy		kg m2	14.6	20%	[11.68; 17.52]
dflex_Izz		kg m2	306	20%	[244.8; 367.2]
dCoGx	SC centre of mass	m	1.57	0.03 m	[1.54; 1.6]
dCoGy		m	0.006	0.03 m	[-0.024; 0.036]
dCoGz		m	-0.04	0.03 m	[-0.07; -0.01]
dBR1_sa_1_x	Rotation participation factors	-	36.64	3%	[35.54; 37.74]
dBR1_sa_2_y		-	0.06	3%	[0.0582; 0.0618]
dBR1_sa_2_z		-	-37.01	3%	[-38.12; -35.89]
dBR1_sa_3_x		-	3.16	3%	[3.0652; 3.2548]
dBR1_sa_3_y		-	0.23	3%	[0.2231; 0.2369]
dBR1_sa_3_z		-	0.03	3%	[0.0291; 0.0309]
dBR1_sa_4_x		-	-0.19	3%	[-0.19; -0.18]
dBR1_sa_4_y		-	3.74	3%	[3.6278; 3.8522]
dBR1_sa_4_z		-	0.02	3%	[0.0194; 0.0206]
dBT1_sa_1_y	Translation participation factors	-	-0.01	3%	[-0.0103; -0.0097]
dBT1_sa_1_z		-	6.54	3%	[6.3438; 6.7362]
dBT1_sa_2_x		-	6.79	3%	[6.5863; 6.9937]
dBT1_sa_3_y		-	-0.04	3%	[-0.0412; -0.0388]
dBT1_sa_3_z		-	3.11	3%	[3.0167; 3.2033]
dBT1_sa_4_z		-	-0.18	3%	[-0.1746; -0.1854]
domega_sa1	SA frequency	rad/s	0.903	20%	[0.7224; 1.0835]
domega_sa2		rad/s	4.419	20%	[3.5352; 5.3028]
domega_sa3		rad/s	6.767	20%	[5.4133; 8.12]
domega_sa4		rad/s	6.92	20%	[5.5392; 8.3088]
ddamp_sa1		-	0.003	0.002	[0.001; 0.005]
ddamp_sa2		-	1.003	0.003	[0.001; 0.005]

$ddamp\_sa3$	SA damping	-	2.003	0.004	[0.001; 0.005]
$ddamp\_sa4$		-	3.003	0.005	[0.001; 0.005]
$dslosh\_sm\_k$	Slosh stiffness coefficient	-	2.698	0.1349	[2.5637; 2.8336]
$dslosh\_sm\_k\_2$		-	4.198	0.2099	[3.988; 4.4078]
$dslosh\_sm\_c$	Slosh damping coefficient	-	0.01	0.0005	[0.0095; 0.0105]
$dslosh\_sm\_c\_2$		-	0.01	0.0005	[0.0095; 0.0105]
$dslosh\_sm\_mass$	Slosh mass	kg	180	2 kg	[178; 182]
$dslosh\_sm\_mass\_2$		kg	280	3 kg	[277; 283]
$dacc1\_Misalign\_x$	Accelerometer misalignment angles	rad	0	0.0005	[-0.0005; 0.0005]
$dacc1\_Misalign\_y$		rad	0	0.0005	[-0.0005; 0.0005]
$dacc1\_Misalign\_z$		rad	0	0.0005	[-0.0005; 0.0005]
$dacc2\_Misalign\_x$		rad	0	0.0005	[-0.0005; 0.0005]
$dacc2\_Misalign\_y$		rad	0	0.0005	[-0.0005; 0.0005]
$dacc2\_Misalign\_z$		rad	0	0.0005	[-0.0005; 0.0005]
$dstr1\_x$	Star tracker misalignment angles	rad	0	0.000262	[-0.00026; 0.00026]
$dstr1\_y$		rad	0	0.000262	[-0.00026; 0.00026]
$dstr1\_z$		rad	0	0.000262	[-0.00026; 0.00026]
$dstr2\_x$		rad	0	0.000262	[-0.00026; 0.00026]
$dstr2\_y$		rad	0	0.000262	[-0.00026; 0.00026]
$dstr2\_z$		rad	0	0.000262	[-0.00026; 0.00026]
$dgyr1\_x$	Gyroscope misalignment angles	rad	0	0.0005	[-0.0005; 0.0005]
$dgyr1\_y$		rad	0	0.0005	[-0.0005; 0.0005]
$dgyr1\_z$		rad	0	0.0005	[-0.0005; 0.0005]
$dgyr2\_x$		rad	0	0.000524	[-0.0005; 0.0005]
$dgyr2\_y$		rad	0	0.0005	[-0.0005; 0.0005]
$dgyr2\_z$		rad	0	0.0005	[-0.0005; 0.0005]
$25 \times dTHR\_x$	Thruster misalignment angles	rad	0	0.0052	[-0.0052; 0.0052]
$25 \times dTHR\_y$	Thruster misalignment angles	rad	1	0.0053	[-0.0052; 0.0052]
$25 \times dTHR\_bias$	Thruster magnitude bias	-	0	0.03	[-3; 3]
$25 \times dTHR\_disp\_x$	Thrusters displacement error	m	0	0.003	[-0.003; 0.003]
$25XdTHR1\_disp\_y$		m	0	0.003	[-0.003; 0.003]
$25XdTHR1\_disp\_z$		m	0	0.003	[-0.003; 0.003]
$dRW\_misalign\_x$	Reaction wheels mounting angles	deg	0	0.3	[-0.3; 0.3]
$dRW\_misalign\_y$		deg	0	0.3	[-0.3; 0.3]
$dRW\_misalign\_z$		deg	0	0.3	[-0.3; 0.3]
$dcss\_bias\_alpha$	CSS bias angle	rad	0	0.00026	[-0.00026; 0.00026]
$dcss\_bias\_beta$		rad	0	1.00026	[-0.00026; 0.00026]
$ddop\_bias\_orb$	OS random bias	-	0	0.333	[-0.333; 0.333]
$ddop\_bias\_os$		-	0	0.333	[-0.333; 0.333]
$ddop\_aging\_orb$	OS aging random bias		0	0.333	[-0.333; 0.333]
$ddop\_aging\_os$		-	0	0.333	[-0.333; 0.333]

4.6 Conclusions

This chapter presents the modelling of the rendezvous between the earth return vehicle (chaser spacecraft) and the orbiting sample in the MSR mission. Some preliminaries

about the mission, frames of reference and architecture of the mission were first discussed. The chaser spacecraft simplified models and their LFT formulations were next presented, with the dynamics of the spacecraft, accelerometer, star tracker (with a time delay included) and reaction wheel LFT models as examples. Validation of the LFTs was then discussed for both dynamical and static gain models. These LFT models are primarily used for control synthesis, and are deliverables to the project consortium. The simplified rendezvous model considered in the robust performance analysis in Chapter 5 does not include LFTs, and is created by linearising a Simulink uncertain block diagram.

The real world high-fidelity rendezvous model framework used in the study was then described. It consists of a complex set of interconnected models, and includes all the systems and subsystems involved in the rendezvous process. The simplified rendezvous model and the high-fidelity model are analysed using robust performance techniques in the following chapter.

Chapter 5

Performance Analysis of the Autonomous Rendezvous System

In this Chapter, the robust performance analysis of the autonomous rendezvous models developed in Chapter 4 is detailed. Two stages of analysis are considered in the study. First, a rigorous design stage performance analysis is carried out using the CE-based method, alongside optimisation-based methods and $\boldsymbol{\mu}$-analysis. Such an analysis ensures that the controller requirements are satisfactorily met within the control synthesis environment first, before proceeding to costly high-fidelity analyses. Subsequently, the robust performance of the iGNC high-fidelity FES model is assessed using optimisation-based methods in Section 5.5. In both stages of the analysis, the critical Terminal Rendezvous and Docking Control (TRDC) mode is considered.

Typically, the validation can guide controller refinements, which can be conveniently made at the design stage. The objectives of this study are:

- to assess the properties of degraded performance in the closed loop autonomous rendezvous linearised system in the presence of the design uncertainties, viz.,

  - to compute the μ- based robust performance margins,
  - to assess the worst case performance,
  - to ascertain if robust performance is achieved,
  - to estimate the probability associated with the worst case (in order to quantify the conservatism of the computed worst case performance),
  - to identify any uncertain parameters that drive the performance more than others, and
  - to generate the probability profile of performance discussed in Chapter 3, and to assess the probabilistic robust performance of the system.

- to determine the efficacy of CE by comparing its results with those of a mature optimisation method (DE) and a traditional analytical technique (μ-analysis).

- to determine the robust performance of the designed controller on the high-fidelity FES model of the autonomous rendezvous.

The first two items are considered in the sections starting from 5.1, while the third is addressed in Section 5.5. To achieve the first two objectives, the optimisation technique DE, the analytical technique μ-analysis, and the probabilistic technique CE are employed. Theoretical and implementation details of these methods are presented in Chapters 2 and 3. Since the quantities of interest are the worst case performance and the associated probability, it is expected that these chosen techniques complement each other.

This chapter is structured as follows: a brief description of the model and the controller is first presented in section 5.1. The two performance objectives — the sensitivity and complementary sensitivity functions — and the uncertain parameters considered in the analysis are discussed in section 5.2. The results of the analysis process are

then detailed for the sensitivity function in section . Based on findings of the study, section 5.3.2 proposes an integrated template for the analysis of spacecraft systems, particularly those that are linear. This template is followed for the complementary sensitivity function and the results presented in Section 5.3.3. Some interpretations of the results are discussed. This chapter also reports results of an optimisation-based performance analysis conducted on the high fidelity model, and states some of the challenges involved in the analysis of a model of this scale.

5.1 iGNC Model Description

Prior to employing the analysis techniques, it is necessary to understand the plant and controller to be analysed, the uncertain parameters involved, and the performance functions. As is standard in $V\&V$, only preliminary details of the plant and controller are provided, since they constitute "black boxes" for the purposes of analysis [116]. Since $V\&V$ traditionally requires there to be a clear separation between the model synthesis and analysis activities, the treatment is superficial. The simplified autonomous rendezvous closed loop system is shown in Figure 5.1. The plant $G(s)$ consists of the chaser dynamics. Its outputs Y consists of a 3 dimensional attitude vector α and a 3 dimensional position vector x. The term Y_m are the measured outputs measured by the sensors i.e., the estimated attitude and position. The signal vector R constitutes the reference forces and torques to be followed, and is computed by two separate guidance algorithms for attitude and translation. The guidance loop typically does not include the effects of perturbations. The controller $K(s)$ computes the error between the reference and the measured signal, and generates a control command U consisting of control torques and control forces. Noise signals enter the closed loop system, usually via sensor measurements. Effects of external disturbance forces and torques are also included in the loop. The designed $K(s)$ needs to to ensure good performance in the presence

of these disturbances, noise signals and in the presence of modelling uncertainties.

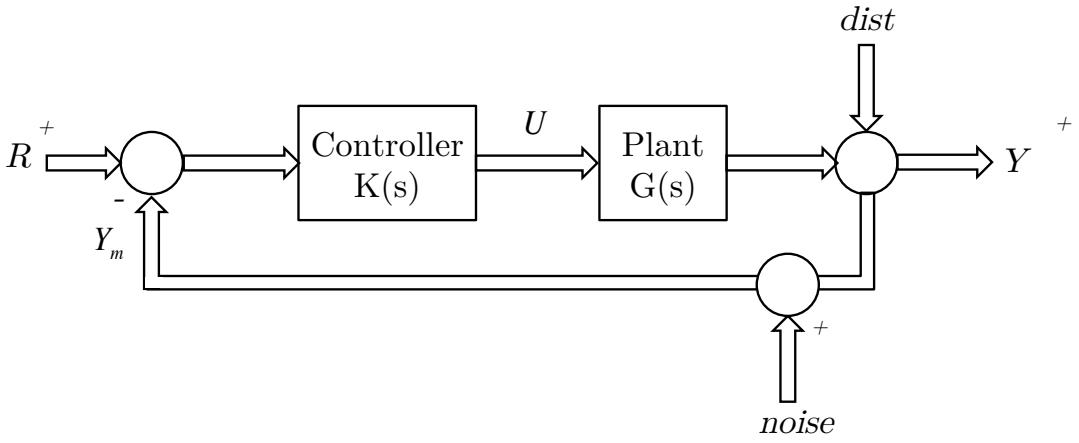


Figure 5.1: Block diagram showing the linearised autonomous rendezvous closed loop system

5.1.1 Linearised Rendezvous Model

The details of the MSR mission description and the development of the dynamical models of autonomous rendezvous and docking are presented in Chapter 4. Several of the dynamical systems involved in the RvD problem are inherently nonlinear. These are to be linearised in order for robust controllers to be developed using principles of linear control theory. The linearisation of these nonlinear systems is performed by industry partners GMV in an informed manner, such that the effects of the dynamics are preserved, but the equations are linear and suitable for linear systems synthesis and analysis. The general methodology employed by GMV is shown in Figure 5.2.

The resulting linearised models are interconnected to yield the plant, which is a 6 degree of freedom coupled dynamics with flexible modes and fuel sloshing modes included. There exists a coupling between the position and attitude due to the effects of flexible modes and fuel sloshing modes. In addition, there exists a kinematics coupling which becomes particularly relevant during the TRDC mode, when the spacecraft size is no

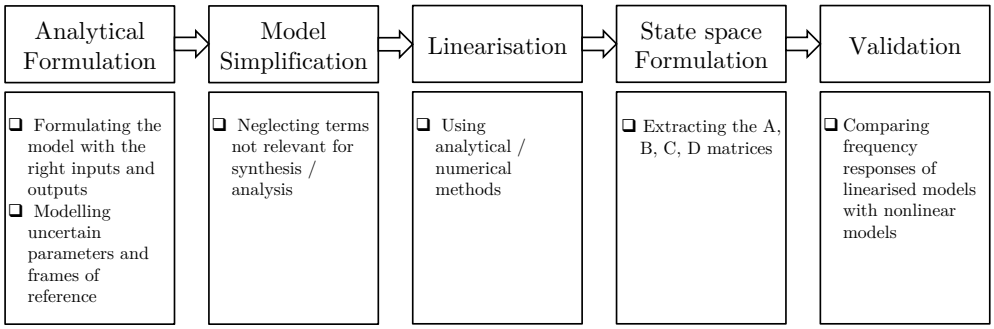


Figure 5.2: Procedure for linearisation of the iGNC model followed by GMV

longer negligible with respect to the distance from the Target. The resulting plant is an uncertain 6-input, 6-output linear system with 69 states and 14 uncertainties. The inputs are the 3×1 torque vector in the chaser body frame T_C, and the 3×1 force vector in the target orbital frame F_T. The outputs of $G(s)$ are the 3×1 estimated chaser attitude vector expressed in the inertial frame (J2000 earth centred inertial frame) α_C, and the 3×1 relative position expressed in the target orbital frame x_{CT}. The singular value plot of the plant w.r.t. frequency is shown in Figure 5.3. The frequency region from about $10^{-1.5}$ to $10^{1.5}$ rad/s reflects problematic regions, possibly attributable to flexible mode and fuel sloshing dynamics. Of particular importance are the peaks above $0dB$, at approximately 10^{-1} rad/s, as they reflect system gains greater than 1 in absolute terms. The effect of uncertainties may be particularly detrimental in these frequency regions, and they must therefore be studied carefully. The nominal plant's poles and zeros are shown in Figure 5.4. All the poles are on the left hand side of the imaginary line, and so the nominal system is stable. Note that some of the poles lie close to the imaginary axis, which could mean that certain disturbances or uncertain parameter values may cause the poles to shift to the right hand side, implying instability. The SISO gain margin is just 2.46dB in the 5th input to 5th output direction, and 3.13dB from the 6th input to the 6th output, further reflecting this issue. An effective robust controller should not only improve stability margins for

the nominal plant, but should ensure that performance is guaranteed even under the influence of uncertainty.

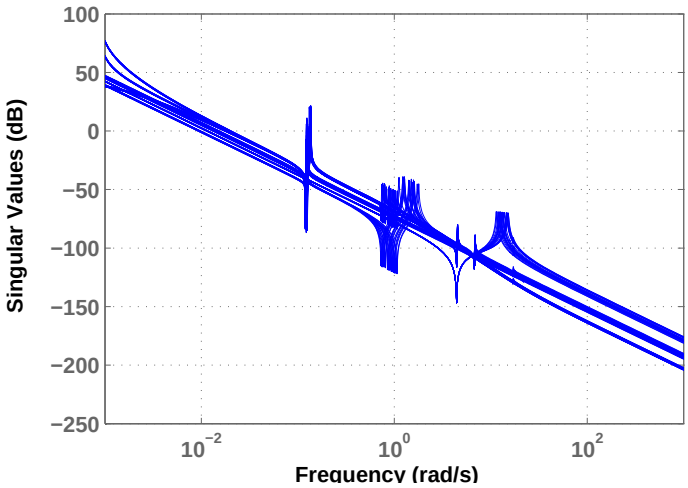


Figure 5.3: Singular values of the linearised plant

5.1.2 Controller

The controller $K(s)$ considered is for the critical TRDC mode, which was made available by TASF. The TRDC mode is activated during the final forced control leading up to docking of the chaser spacecraft with the sampling cannister. $K(s)$ is a thruster based, 6 degree of freedom forced control for both the position and attitude. It is a H_∞ design in a single input single output (SISO) structure, with three channels each for translation and attitude. The singular values of the controller are shown in Figure 5.5.

The controller design does not account for attitude-position coupling due to propellant motion in the tanks. The controller requirements for the TRDC mode are stringent, and are expressed as follows:

- robustness to uncertainty on chaser spacecraft parameters (centre of mass, iner-

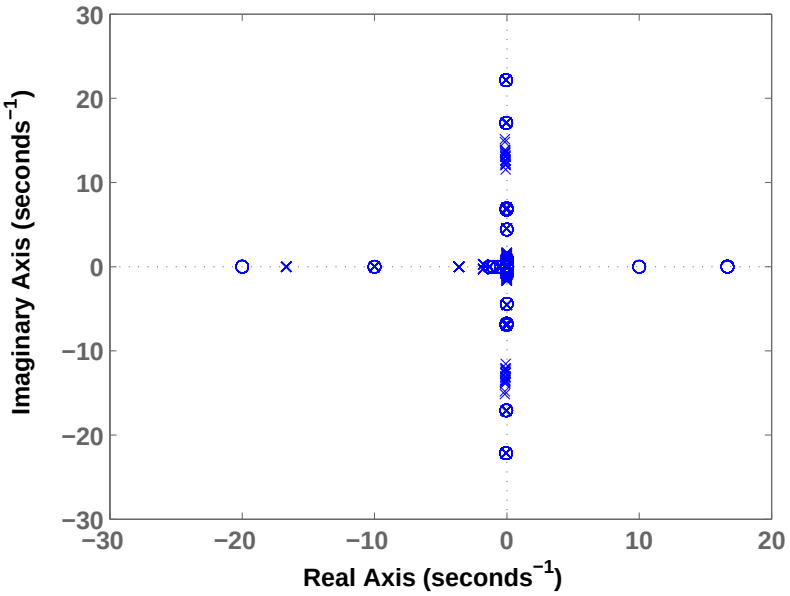


Figure 5.4: Pole-zero map of the linearised plant

tia),

- robustness to effects of flexible appendages and fuel sloshing on attitude and translation, and
- attenuation of exogenous disturbance.

As stated in chapter 2, the focus is on robust performance requirements. Although stability is critical, often performance degradation occurs before the onset of instability. Thus it is vital to study the degradation of performance, and the uncertain parameter values responsible for it.

5.1.3 Uncertain Parameters

This section lists the design uncertainties considered in the controller synthesis. These will be considered in the simplified analysis, thus verifying that the controller achieves the objectives it was designed to. These uncertain parameters are shown in Table

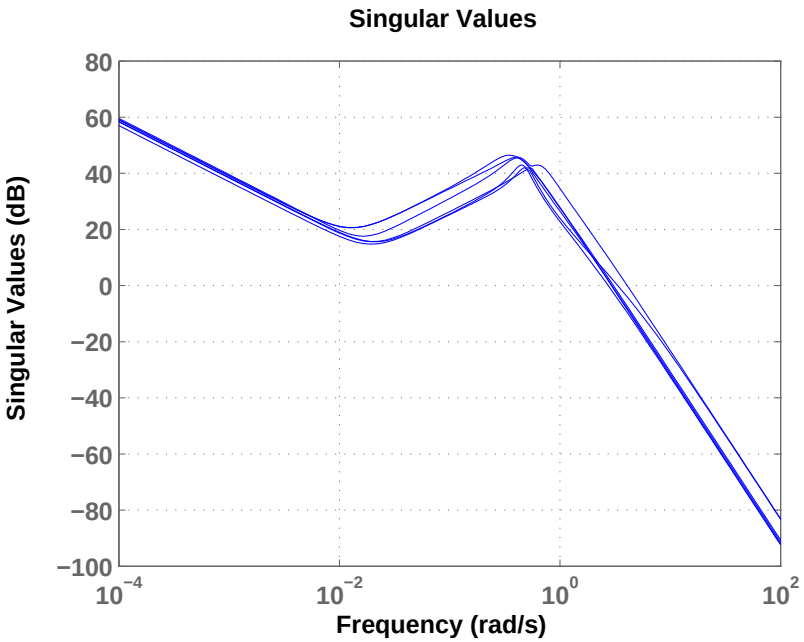


Figure 5.5: Singular values of the controller

5.1, along with their names and physical meaning. In the interest of readability, the parameters are not referred to by their names in the sequel. Instead, δ_j is used, where j indexes the parameter.

In this work, CE and DE implementations consider the parameters to be distributed according to a truncated Gaussian distribution with a mean 0, and a standard deviation 0.33. Such a selection ensures that the 3-σ values of the normal distribution fall at the [+1,-1] truncation bounds. This ensures that the parameter space in [+1,-1] is sufficiently explored, yet the space is largely Gaussian in nature.

5.2 Performance Objectives

The performance objectives are chosen such that they encapsulate the TRDC controller objectives enumerated in section 5.1. Disturbance rejection and reference tracking in the frequency domain are associated with the sensitivity function S and the comple-

Table 5.1: iGNC design uncertain parameters

Uncertain Parameter	**Parameter Name**	**Description**
δ_1	$dBR\_sa\_y\_mode\_1$	Rotational participation along $y-$axis
δ_2	$dBT\_sa\_x\_mode\_1$	Translational participation along $x-$axis
δ_3	$dIxx\_SA$	Inertial participation along $x-$axis due to solar arrays
δ_4	$dIxx\_body$	Inertia of central body along $x-$axis
δ_5	$dIyy\_SA$	Inertial participation along $y-$axis due to solar arrays
δ_6	$dIyy\_body$	Inertia of central body along $y-$axis
δ_7	$dIzz\_SA$	Inertial participation along $z-$axis due to solar arrays
δ_8	$dIzz\_body$	Inertia of central body along $z-$axis
δ_9	$dfreq\_sa\_mode\_1$	Natural frequency of first mode of solar array
δ_{10}	$dmass$	Mass of the spacecraft
δ_{11}	$dslosh\_mass\_sm1$	Fuel slosh mass in tank 1
δ_{12}	$dslosh\_mass\_sm2$	Fuel slosh mass in tank 2
δ_{13}	$dslosh\_stiffness\_sm1$	Fuel slosh stiffness constant in tank 1
δ_{14}	$dslosh\_stiffness\_sm2$	Fuel slosh stiffness constant in tank 2

mentary sensitivity function T [15], as already stated in Section 2.3. The sensitivity function (sometimes also called the sensitivity matrix)[24] is the transfer function from the disturbance to the output, given by $S = (I + G(s)K(s))^{-1}$, and the complementary sensitivity function is the transfer function from the noise to the output $T = G(s)K(s)(I + G(s)K(s))^{-1}$. In the MIMO sense, the maximum singular value of the sensitivity function describes the disturbance rejection ability of the closed loop system (recall that the maximum singular value is a generalisation of SISO gains, and gives the maximum gain in any input direction [15]). This measure needs to be assessed in the presence of uncertainty. Note that disturbance rejection is particularly crucial when the control is thruster based (such as in the TRDC control mode), since such controllers are more subject to thrust magnitude and direction errors (due to disturbance torques from the environment).

The criteria are defined by the iGNC project industrial consortium to be $\bar{S}_v(S) < 6dB$ and $\bar{S}_v(T) < 6dB$. The criteria are in keeping with the industry standard, and corresponds to a gain that is approximately 2 (interpreted as a doubling in magnitude from disturbance to outputs). The performance criterion is formally stated as

$J = \bar{S}_v(S) < 6dB$. A probabilistic performance criterion is also formulated, as in Equation (2.23). The criterion is $P[J \geq 6dB] < 10^{-6}$.

5.3 Results of the Analysis

Having defined the performance function $J(\delta)$ and the uncertain parameters δ to be considered, what follows is an account of the performance analysis procedure and the results obtained. Each result eventually builds towards an integrated approach to performance analysis of the linearised iGNC system. The procedure consists of implementing algorithms described in chapters 2 and 3. All implementations are done in Matlab version 8.2 on a 64-bit Intel Core-i7 computer with 8 gigabytes of RAM.

5.3.1 Sensitivity Function

The verification process for the sensitivity function S is commenced by observing its singular value plot w.r.t frequency (obtained using the Matlab function `sigma`) shown in Figure 5.6. The blue lines show all the singular value responses obtained, while the green dashed line shows the response corresponding to the nominal S function, whose peak singular value is $\bar{S}_v(S(\delta)) = 4.658dB$ at $0.1528rad/s$, and the peak singular value over 100 random samples of S is $5.423dB$, which occurs at $0.1239rad/s$. It is observed here that the frequencies lying in the range $10^{-1} - 10^{-1.2}$ are particularly important, as indicated by Figure 5.6, since the response in this region is close to the red $6dB$ line. Figure 5.7 shows this frequency range in further detail, where it may be observed that although there is a distinct peak at $0.1236rad/s$, other frequency points in the vicinity also have high responses. This is important because certain combinations of uncertainty parameters can cause the response to grow high in the regions to the right

of the observed peak. It is noteworthy that this region corresponds directly to the problematic frequency region in the open loop plant shown in Figure 5.3, with singular value peaks above $0dB$.

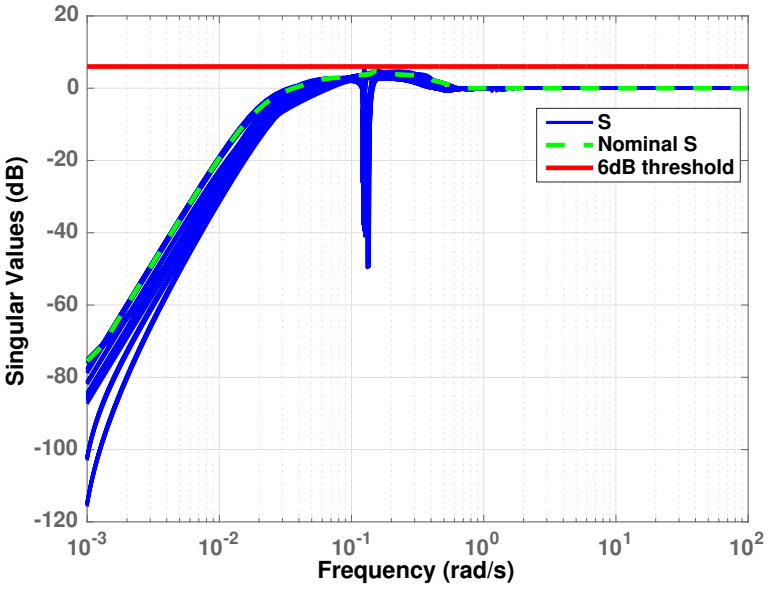


Figure 5.6: Singular values of the sensitivity function

Having had a preliminary understanding of the performance function behaviour, it is desired to rigorously determine its worst case value, and to check if the 6dB threshold is respected for all combinations of the uncertain parameters. With this objective, an optimisation-based analysis is set up using the DE algorithm available in the Worst Case Analysis Toolbox - II (WCAT-II), which has been successfully used to in worst case analysis [68]. DE is observed to have excellent capabilities in performance analysis of spacecraft systems [11] [18] [72]. The principle and working of DE are discussed in chapter 2. The DE run is set up with the algorithmic parameters listed in Table 5.2, which are originally suggested in [75]. With the chosen population size and maximum iterations allowed, the total number of evaluations of the performance function

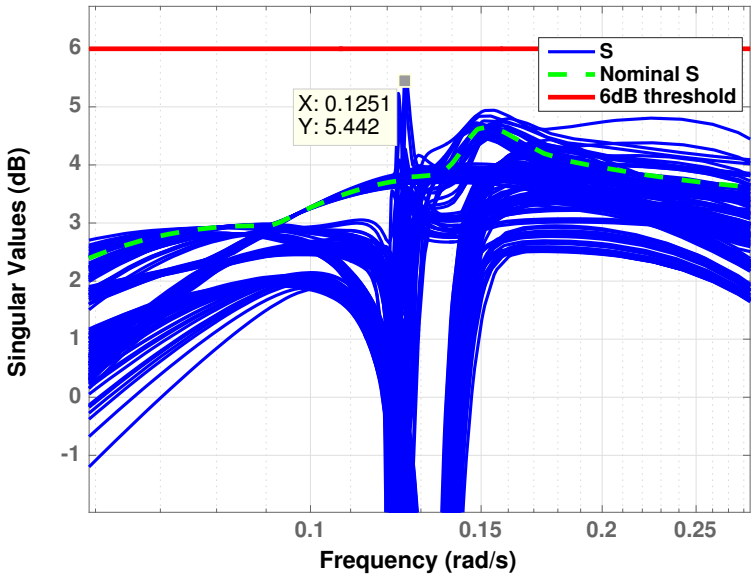


Figure 5.7: View showing peak singular values of the sensitivity function

Table 5.2: DE algorithmic parameter values used in the analysis

DE Parameter	Parameter Value
Population size	25
Maximum iterations	100
Crossover ratio	0.8
Step size	0.8

is $100 \times 25 = 2500$, while a single performance evaluation takes approximately $0.239s$. Therefore, the total time taken for the run is about 10 minutes.

This DE optimisation run resulted in a worst case performance value $J^*_{DE} = 5.6413dB$. This value, higher than the one obtained from the singular value plot but still lower than the $6dB$ threshold, shows that the designed controller achieves robust performance in terms of disturbance rejection to the design uncertainties. The worst case uncertain parameter combination Δ^*_{DE} obtained is shown in Table 5.5. Note that the worst case parameter combination includes values pushed towards the extremes [-1,1], e.g. param-

eters δ_{11} and δ_{12}. However it also includes parameters that lie towards the middle of the parameter space, e.g. δ_6. This supports the reasoning that rigorous verification and validation approaches are necessary, since examining only combinations of the extreme values of the uncertainties (such as in vertex analysis [117] [56]) would not give an accurate solution.

The evolution of the performance values as the DE iterations progress is shown in Figure 5.8. The solid red lines show the maximum cost values obtained among the population in each iteration, while the blue dashed lines show the mean values of the performances obtained in the population. It is observed empirically that a close convergence between the mean value plot and the maximum value plot indicates that the optimisation solution has reached close to the true maximum. In addition, the jumps in the maximum cost reduce to relatively small increments after the first 300 evaluations, which also suggests that the optimisation has reached a good estimate of the true maximum, beyond which there is only minor improvement possible. Therefore, the analysis strongly indicates that the worst case performance is smaller than $6dB$ in the presence of the design uncertainties. Proceeding further in the analysis, it is now desired to find the probability associated with the performance, particularly the worst case performance value. A CE-based analysis is carried out. The concepts of probability profile of performance and the CE algorithm are introduced in Chapter 3. The algorithmic parameters used in the run are shown in Table 5.3, and CE is run in the adaptive mode. A large number of maximum CE iterations t_{max} is allowed in this case as each performance function evaluation takes only 0.2 seconds. Recall that CE by default evaluates the performance function $N + t * N_1$ times, where t is the number of iterations taken to arrive at an optimal estimate of the mean and standard deviation. Thus, at every value of γ, the minimum allowed evaluations is 1500 (with $t = 1$) and a maximum of 16000 (with $t = 30$). The rarity factor is set at $\rho = 0.1$,

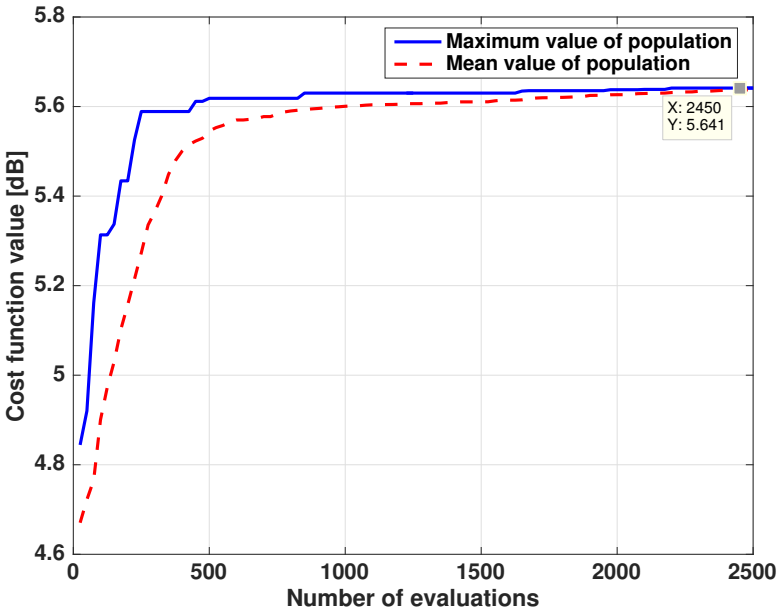


Figure 5.8: Results of the DE optimisation run: evolution of performance values with the iteration count

so that $t > 1$ only for those performances that have a probability of occurrence lower than 0.1 The threshold standard deviation value means that the standard deviation of each uncertain parameter is at most 10^{-6}, ensuring that a very tight IS distribution is arrived at, particularly for rare events.

The performance levels γ at which the probability is to be estimated are chosen from $4.2dB$ to $5.7dB$. A two-level step size strategy is adopted, with a step-size of $0.1dB$ from $4.2dB$ to $5.4dB$, and a finer step-size of $0.05dB$ thereafter in the more critical performance range (i.e., performances close to the worst case found by DE). Recall that the nominal performance value is $4.66dB$, however, in this system it has been observed that with certain combinations of uncertain parameters, the performance function value reduces from the nominal value γ_{nom}. Hence, the initial performance level is set below γ_{nom}, at $4.2dB$.

Table 5.3: CE Algorithmic Parameter Values

CE Parameter	**Parameter Description**	**Parameter Value**
N	Evaluations at each CE cycle	500
N_1	Final IS run evaluations	1000
ρ	Rarity factor	0.1
t_{max}	Maximum number of CE iterations allowed	30
σ^*	Threshold standard deviation	10^{-6}

Table 5.4: Probabilities associated with various performance levels in the linearised iGNC model.

γ **[dB]**	$P[J(\delta) \geq \gamma]$	**Evaluations**
4.3	1	1500
4.4	1	1500
4.5	0.983	1500
4.6	0.771	1500
4.7	0.394	1500
4.8	0.119	2000
4.9	0.031	1500
5	0.019	1500
5.1	0.010	2000
5.2	0.007	1500
5.3	0.004	1500
5.4	0.001	1500
5.45	$2.837*10^{-3}$	1500
5.5	$3.723*10^{-4}$	2000
5.55	$8.046*10^{-6}$	2000
5.6	$7.822*10^{-09}$	2500
5.65	0	16000
5.7	0	1500

Remark: In this section, the terminology is modified, to enable convenient and concise discussion of the results. The estimated probability found by CE is simply referred to as the probability. The author assumes that the non-deterministic sense of the technique has been established to a sufficient extent in Chapter 3.

Figure 5.9 shows the probability profile of performance generated by CE, i.e., $P[J(\delta) \geq \gamma]$ plotted against γ. The precise values of probability associated with each performance level are shown in Figure 5.4. The following interpretations are now derived:

- The probability remains 1 for γ values up to 4.4 dB. This implies that the peak

value of singular value is always greater than $4.4dB$, which corresponds to an absolute gain of about 1.66 from the disturbance to the plant output.

- As γ increases, the associated probability steadily falls from 0.983 at $\gamma = 4.5dB$ to 0 at $\gamma = 5.65dB$, implying that the maximum possible performance does not exceed 5.65 dB. This supports the result found by DE, with a worst case value of $J^*_{DE} = 5.6442dB$.

- Interestingly, the probability associated with the nominal performance value $4.66dB$ is not 1 as expected, showing that some uncertain parameter combinations may in fact reduce performance from the nominal performance.

- The probability associated with $\gamma = 5.6$ is of the order 10^{-9}, which roughly is a one in one billion likelihood that a performance greater than 5.6 is achieved. This implies that the worst case performance lies between $5.6dB$ and $5.65dB$, which is indeed confirmed by the worst case found by DE.

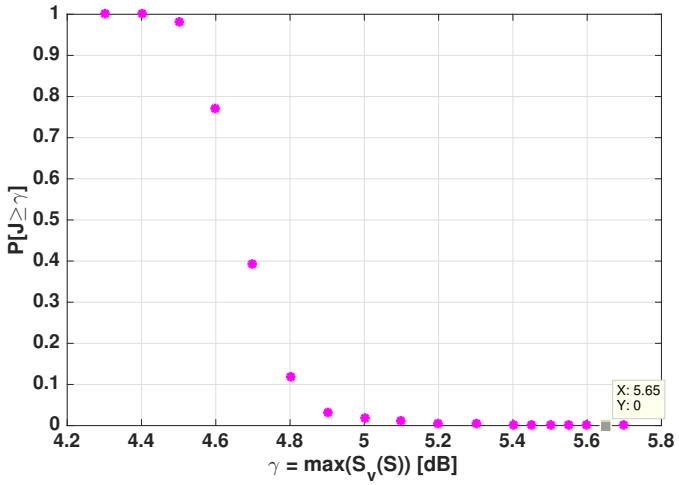


Figure 5.9: Probability profile of performance generated by CE

Before proceeding to a discussion on the insights that may be obtained from the PPoP,

it is important to investigate the performance of the CE-based algorithm over multiple runs. Therefore, the CE algorithm run is repeated ten times, preserving the parameter values in Table 5.3. Any changes in the estimated probability $P[J(\delta) \geq \gamma]$ at a given level of γ are therefore due only to the importance sampling distribution computed by CE (and the inherent randomness associated with sampling from such a distribution).

The estimated probabilities $P[J(\delta) \geq \gamma]$ computed over these ten runs are plotted in the form of errorbars as in Figure 5.10. The largest errorbars (implying the highest variations in the results) correspond to the decay region introduced in Chapter 3. Indeed, the largest deviation occurs at $\gamma = 4.7$dB, with the estimated probability varying in the range [0.365,0.417]. Such variations are expected in the decay region, since the importance sampling distribution found by CE at these γ values has a fairly high standard deviation, due to which the selected samples cover a large area of the parameter space. In the current study, the critical region of the PPoP (referred to in Chapter 3 as the performance values corresponding to probabilities lower than 0.1) assumes importance. Notice that the errorbars diminish in size as γ becomes rarer. For example, the error between the maximum and minimum values found at γ=5.6 is 4.34×10^{-08}. This confirms that CE is highly accurate for estimating probabilities associated with rare-events.

Returning now to discussing the insights that may be gained from the PPoP, Figure 5.9 is revisited. As discussed in Chapter 3, the PPoP can be a useful tool to gain insight into the probabilities associated with various performance levels. However, in the current form of the probability profile of performance, the drawing of such inferences is limited only to the specific performance levels γ considered in the CE analysis, which leaves gaps in the plot. However, it is infeasible to run CE for very small step sizes in reality. Therefore, in order to enhance the capabilities of the PPoP, a curve fitting is done using a Piecewise Cubic Hermite Interpolating Polynomial, also known as 'PCHIP'.

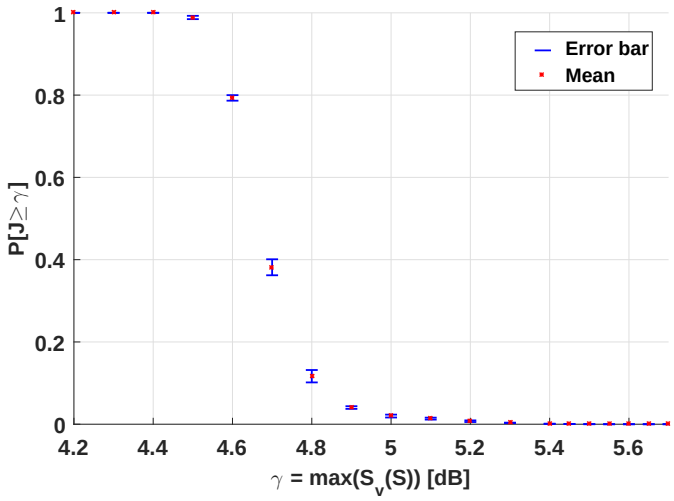


Figure 5.10: Error bars of the probability profile of performance generated by CE over ten runs

Matlab's Curve Fitting Toolbox [118] is utilised for this, and the resulting probability profile of performance fitted curve is shown in Figure 5.11. Such a curve is suitable for making inferences about performance values and the associated probabilities. For instance, the probability associated with $\gamma = 4.75dB$ is 0.26. Conversely, it is possible to infer the performance associated with a given probability. For example, a probability of 0.3 is associated with a performance of $4.735dB$. In other words, roughly 30% of the sampled parameters result in a performance value greater than $4.735dB$). This affords an intuitive understanding of a system's performance. Recall that this curve is similar to the PPoP curve predicted in Figure 3.3. Adhering to the terminology, the performances from $4.55dB$ up to $4.8dB$ corresponds to the decay region, while performances beyond $4.8dB$ corresponds to the critical region. Since all of these regions are well within the $6dB$ bound, it is clear that both deterministic and probabilistic robust performance is guaranteed.

The final importance sampling run evaluates the S performance function $N_1 = 1000$

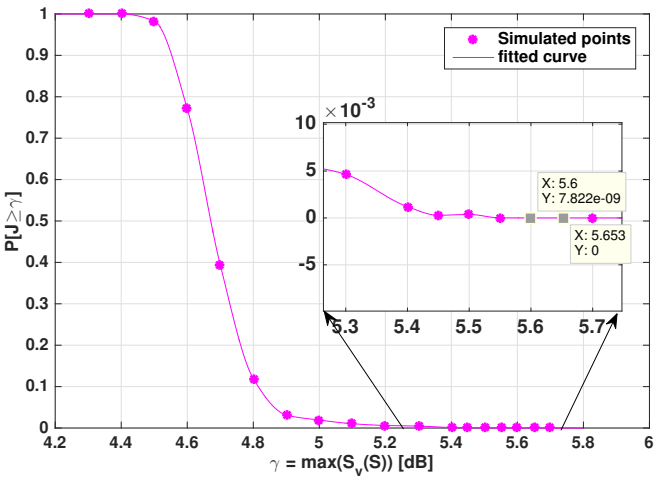


Figure 5.11: Curve fit of the probability profile of performance

times, with the normal distribution with mean μ_T and standard deviation σ_T. The performance values obtained at each value of performance level $\gamma$the performance increases as γ increases, while the variance of these performances reduces. This indicates the final IS distribution $f(\mu_T, \sigma_T)$ gets closer to the optimal worst case distribution as γ progresses closer to the worst case performance. At $\gamma = 5.6$ and $\gamma = 5.65$, the cost converges to a value of 5.627dB over all 1000 evaluations, indicating that a very tight optimal distribution is arrived at by CE. This final value is termed the CE-based worst case performance $J^*_{CE} = 5.627dB$. The performance values at each IS run are scattered and shown in Figure 5.12, which shows how CE effectively reduces their variance.

A note on the number of evaluations: at each performance level the CE algorithm only evaluates the performance function 1500-2500 times, due to the adaptive mode of CE. It is only at $\gamma = 5.65dB$ that the maximum allowed evaluations (500*30+1000) are carried out, as the mean and standard are refined aggressively to seek an IS distribution that would obtain a performance value greater than $5.65dB$. Since such a distribution does not exist (at least within a standard deviation of 10^{-6}), the final probability of

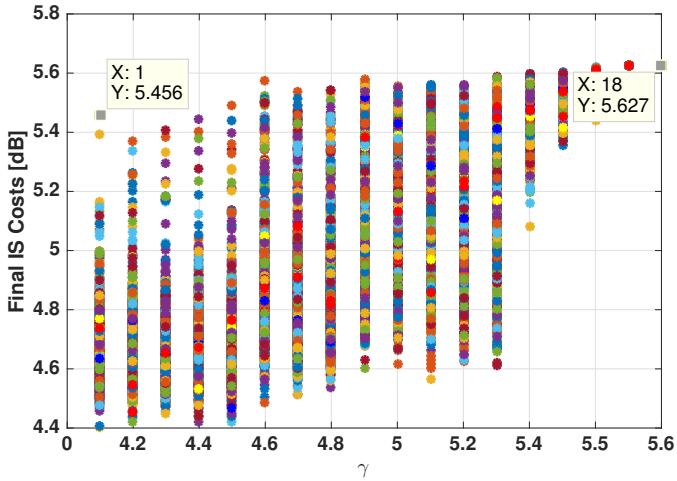


Figure 5.12: Performance values found at each CE performance level

zero was obtained with a high confidence. The current run took nearly 2 hours considering all γ values, which is reasonably low. If the system under consideration is computationally intensive, it is recommended for t_{max} to be lower. Generally speaking, this is the exploration vs exploitation issue faced by most optimisation methods. A preliminary understanding of the problem complexity can guide t_{max} selection.

Having gained information on the performance, it is of interest to understand the uncertain parameter regions associated with poor performance. Figure 5.13 shows the evolution of the final mean and standard deviation values arrived at by CE as γ increases. Recall that these mean and standard deviation values (μ_T and σ_T respectively) specify the distribution used for the subsequent IS run. Therefore, they are indicators of the uncertain parameter regions which yield various performance levels γ. It is apparent that towards the higher performance values, the standard deviations σ_T reduce to very small values (necessarily lower than $\sigma^* = 10^{-6}$, as set). Thus, the IS distributions are extremely narrow, with a large peak at the corresponding mean values. Consequently, μ_T at the worst case performance γ^* may be considered as the CE worst case

parameter values δ^*_{CE} The uncertain parameter values are listed in Table 5.5. Again, note that there are some parameters that are pushed towards ± 1, and others that have intermediate values.

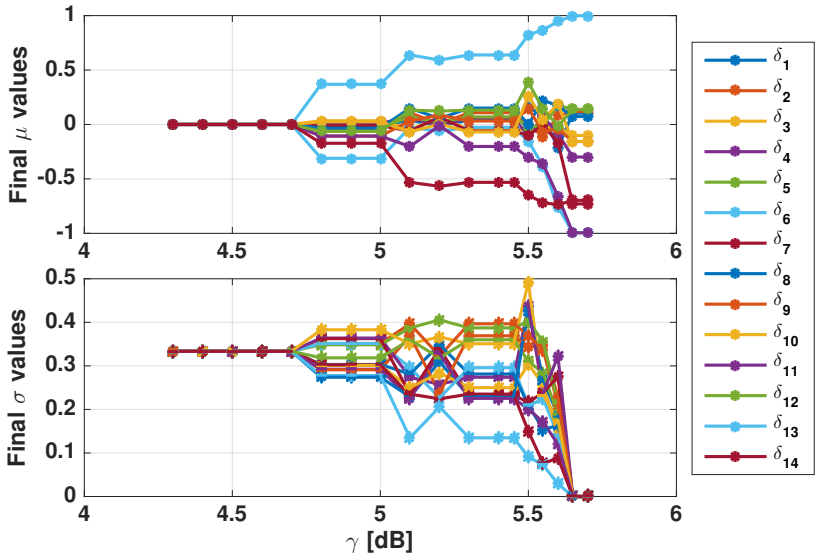


Figure 5.13: Final mean and standard deviation values vs γ

These initial runs of DE and CE provide useful information, which will be revisited shortly. In the next part of the section, the robust performance analysis problem is approached from a more traditional, analytical viewpoint using $\boldsymbol{\mu}$-analysis, which has been alluded to in the previous chapters. Since the sensitivity function of the linearised RvD plant is linear, $\boldsymbol{\mu}$-analysis is readily applicable. The implementation presented here uses $\boldsymbol{\mu}$-analysis to study robust performance, and is performed using Matlab's Robust Control Toolbox. The function `robustperf` is used to carry out the analysis, which computes upper and lower bounds on the robust performance margin [50]. It also provides information on the critical frequency and the uncertain parameter values associated with the worst case performance. A code snippet showing the robust

performance function call is shown in Figure 5.14.

```
freqs = logspace(-2,2,100); %Generate 100 frequency points between 10^-2 and 10^2
S_freqs = ufrd(S,freqs); %Create uncertain frd models at those frequencies
opt=robustperfOptions('Mussv','ag8','Sensitivity','off','Display','on'); %Robust performance options
[StabMarg,Wc_Unc,Report,Info] = robustperf(S_freqs,opt); %Compute robust performance margins using mu
```

Figure 5.14: Code snippet showing μ-analysis based robust performance

In order to make the system S suitable for μ-analysis, the performance objective must be incorporated, which is done by introducing a diagonal scaling matrix $Ws = 1/6dB * I_6$ where I_6 represents the 6×6 Identity matrix. Thus the system is $Ws(I + KG)^{-1}$. Note that the full block complex uncertain matrix often included in robust performance studies is included implicitly in Matlab's `robustperf` function [50].

Results of the μ-analysis are shown in terms of the upper and lower bounds of the robust performance margin in Figure 5.15. The upper bound of the robust performance margin is 1.0754 and the lower bound is 1.006, at a critical frequency of 0.124 rad/s. The robust performance margin indicates the minimum amount of (normalised) uncertainty in the system for performance to degrade beyond the threshold. This would mean that having these margins lower than 1 would cause an uncertain system to degrade beyond the threshold, at smaller uncertainty values. Since in this analysis, both the upper and lower bounds are greater than 1, this effectively guarantees robust performance. The performance margin is guaranteed to be below the upper bound, and above the lower bound performance. In other words, the S loop can tolerate between 100.6% and 107.54% of the modelled uncertainty. The algorithm also returns an uncertain parameter vector associated with poor performance. A model uncertainty of corresponding to 107.54% can lead to a scaled gain of 0.94. This combination of uncertain parameters is listed in Table 5.5. Substituting this uncertainty in the sensitivity function, the worst

case performance is found to be 5.63dB (≈0.94*6dB).

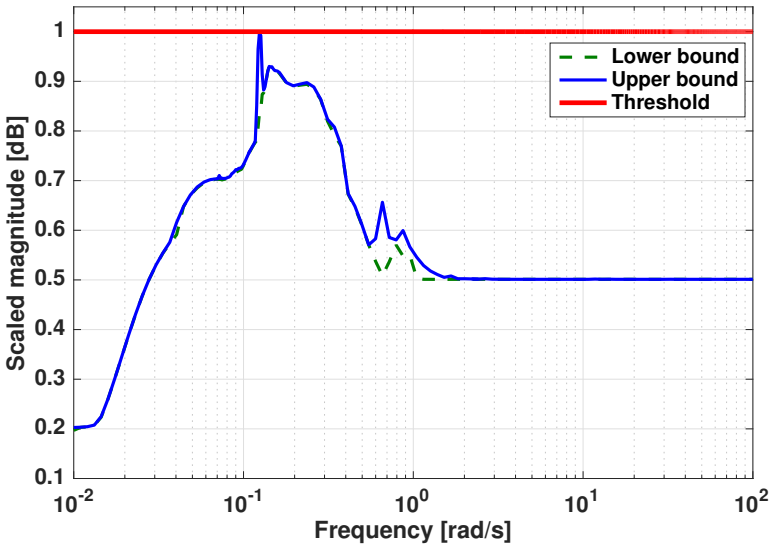


Figure 5.15: Upper and lower bounds found by the slow method of μ-analysis

Understanding how these three powerful techniques complement each other also involves comparing the resulting worst case uncertain parameter values. In addition, this exercise may also provide information on driving uncertainties. Table 5.5 shows the worst case uncertain parameters as given by DE, CE and μ-analysis. These values are shown in the bar plot in Figure 5.16. It is observed that there are striking dissimilarities between the obtained worst case values of several uncertainties in their magnitudes, and even in the directions between the analyses. This could suggest the following possibilities:

- The non-matched uncertain parameters do not impact the performance function greatly.
- Separate regions exist in the parameter space that yield poor performance.
- One or more of the methods is not effective in finding the true worst case param-

Table 5.5: Worst case uncertain parameter values and performances obtained by the DE, CE and μ analyses

Uncertainty	**Uncertainty Name**	δ^*_{DE}	δ^*_{CE}	δ^*_{μ}
δ_1	$dBR\_sa\_y\_mode\_1$	-0.4969	0.1040	0.07969
δ_2	$dBT\_sa\_x\_mode\_1$	-0.5538	0.1204	0.03766
δ_3	$dIxx\_SA$	-0.2638	-0.1007	0.1667
δ_4	$dIxx\_body$	0.6757	-0.2983	-0.0069
δ_5	$dIyy\_SA$	0.6361	0.1378	-0.5730
δ_6	$dIyy\_body$	0.7609	-0.9939	-1.0930
δ_7	$dIzz\_SA$	-0.3128	-0.7293	0.04173
δ_8	$dIzz\_body$	-0.7666	0.07266	0.2581
δ_9	$dfreq\_sa\_mode\_1$	0.8354	-0.1564	-0.1179
δ_{10}	$dmass$	-0.2985	-0.1555	1.0930
δ_{11}	$dslosh\_mass\_sm1$	-0.9716	-0.9954	-1.0930
δ_{12}	$dslosh\_mass\_sm2$	0.9213	0.1379	1.0930
δ_{13}	$dslosh\_stiffness\_sm1$	0.9877	0.9991	1.0930
δ_{14}	$dslosh\_stiffness\_sm2$	-0.8112	-0.6968	-1.0930
Worst case values [dB]		**5.6413**	**5.6266**	**5.631**
Frequency [rad/s]		**0.125**	**0.125**	**0.124**

eter combinations.

Based on the first point, it is postulated that only a subset of the uncertainties drive the performance to a large extent. It may be tempting to conclude prima facie that only the parameters δ_{11}, δ_{12}, δ_{13} and δ_{14} drive the performance the most, since these uncertainties match in their directions across the 3 analysis methods. However, further study is required to accurately determine which of these parameters drive the performance.

A local sensitivity analysis is therefore carried out to assess the impact of each of the design uncertainties on the performance function. The analysis is done by fixing the parameter vector at the worst case value[1]. Then, each parameter is varied one at a

[1]A local sensitivity analysis, i.e., with parameters taken one at a time, is usually insufficient to discover relationships between parameters and the performance function. Global sensitivity analysis is carried out when higher accuracy is needed, and will be considered in future work.

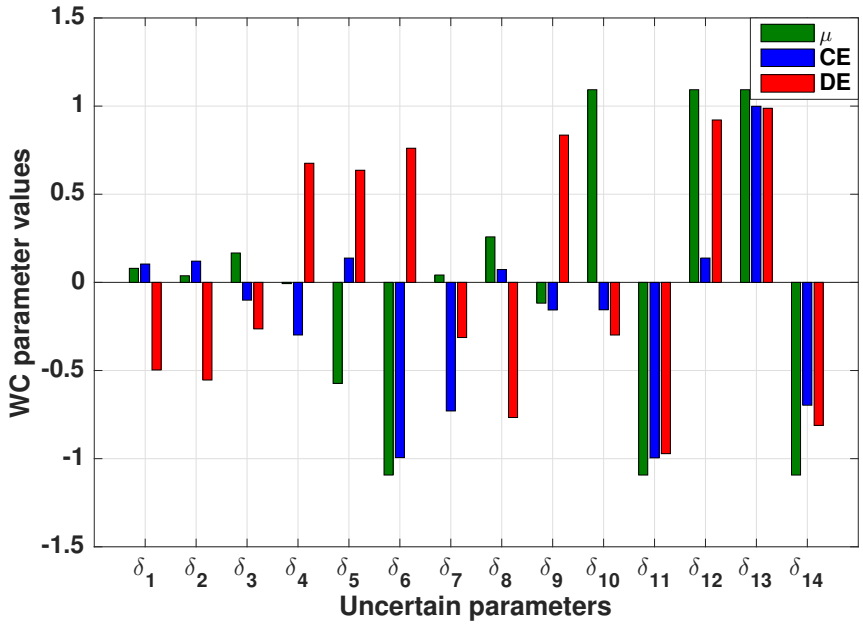


Figure 5.16: Comparison of the worst case parameter values found by DE, CE and μ-analysis.

time from its minimum to its maximum value (i.e., from -1 to +1 in steps of 0.1), and the costs are recorded and plotted. This exercise is carried out for the following cases:

- DE worst case parameter value δ^*_{DE}
- CE worst case parameter value δ^*_{CE}

Table 5.6 presents the results of the sensitivity analysis in terms of the maximum performance (or cost), the minimum cost, and the relative percentage difference between the two, which is calculated as $\frac{|MaximumCost - MinimumCost|}{|MaximumCost|} * 100$. Figure 5.17 shows the trend followed for the DE based sensitivity analysis.

Clearly, some parameters cause a far greater variation in the performance than others. If a parameter causes at least a 1% variation in cost, it is considered to be a *driving parameter*. The blue highlighted rows in the table show the parameters for which the

Table 5.6: Sensitivity analysis results showing driving uncertainties

Parameter	Maximum Cost		Minimum Cost		% Variation	
	DE	CE	DE	CE	DE	CE
δ_1	5.6413	5.6266	5.6412	5.6266	0.0026	$2.75e^{-4}$
δ_2	5.6413	5.6266	5.6413	5.6266	$2.78e^{-4}$	$2.71e^{-5}$
δ_3	5.6420	5.6266	5.6346	5.6266	0.1307	0.0699
δ_4	5.6420	5.6269	5.6411	5.6260	0.0156	0.0174
δ_5	5.6413	5.6269	5.6413	5.6263	$1.14e^{-7}$	0.0107
δ_6	5.6413	5.6267	5.527	5.5433	2.067	3.5602
δ_7	5.6417	5.6268	5.6406	6.6255	0.0204	0.0233
δ_8	5.6419	5.6266	5.525	5.5686	2.1064	1.0414
δ_9	5.6414	5.6266	5.6390	5.6263	0.0421	0.0050
δ_{10}	5.6414	5.6268	5.6319	5.6186	0.1686	0.1774
δ_{11}	5.6419	5.6267	5.4801	5.4016	2.9519	4.1674
δ_{12}	5.6413	5.6265	5.5485	5.5641	1.6725	1.1212
δ_{13}	5.6425	5.6267	4.4852	4.8135	25.8006	16.8922
δ_{14}	5.6412	5.6265	4.6898	5.1173	20.2872	9.9515

Table 5.7: Driving uncertain parameters for the sensitivity function

Driving Uncertainty	Uncertainty	Uncertainty Name
δ_1^S	δ_6	$dIyy\_body$
δ_2^S	δ_8	$dIzz\_body$
δ_3^S	δ_{11}	$dslosh\_mass\_sm1$
δ_4^S	δ_{12}	$dslosh\_mass\_sm2$
δ_5^S	δ_{13}	$dslosh\_stiffness\_sm1$
δ_6^S	δ_{14}	$dslosh\_stiffness\_sm2$

% variation is 1% in both CE and DE. It is interesting to note that these do not exactly match the initial hypothesis, since δ_6 and δ_8 also drive the performance in addition to δ_{11}, δ_{12}, δ_{13} and δ_{14}. Table 5.7 lists these driving uncertainties. The driving uncertainties are represented as δ_1^S-δ_6^S, where the superscript S, stands for the sensitivity function. It is not necessary for the driving uncertainty set δ^S to drive other cost functions.

The driving uncertainties obtained are sensible from a physical perspective in the system, since the sloshing dynamics, which constitute 4 of the 6 driving uncertainties, have the most impact on the sensitivity function. It is also noteworthy that the parameters δ_{13} and δ_{14}, which are the slosh stiffness parameters, drive the performance the most, as evidenced by Figure 5.17 and Table 5.6. This implies that under high effects

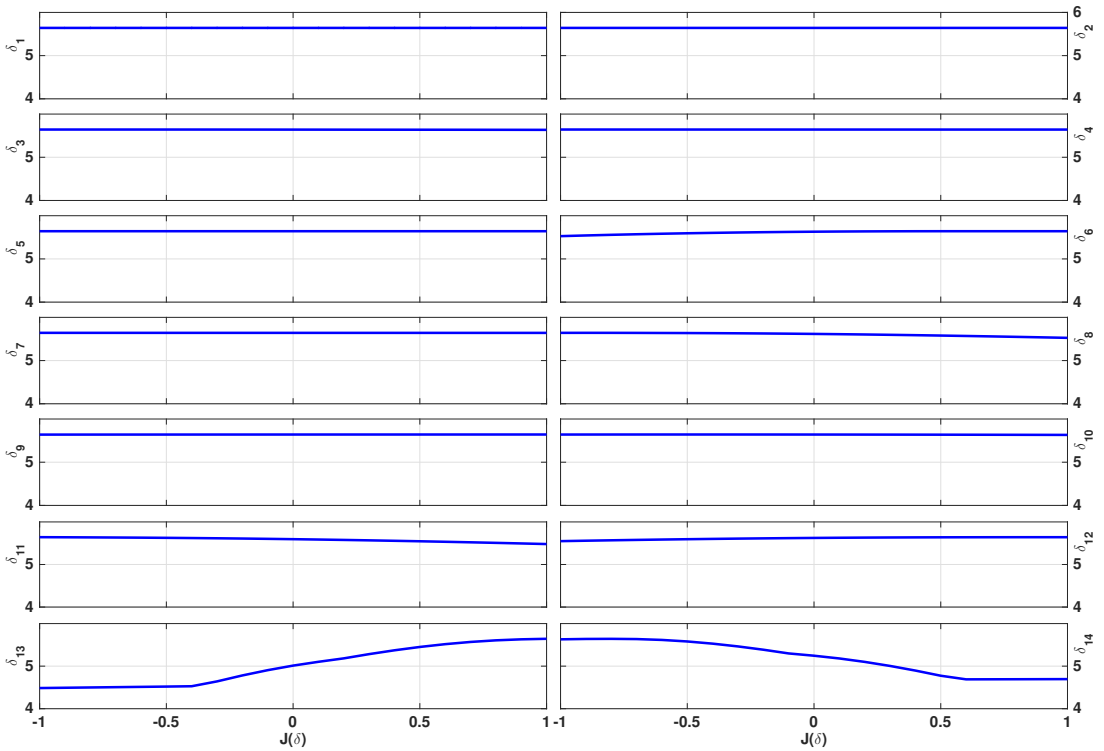


Figure 5.17: Sensitivity analysis with DE worst case values

of sloshing, the gain from disturbance to plant input is maximised. Interestingly, the behaviour of δ_{13} and δ_{14} appear to be in opposite directions, i.e., the stiffness parameter for both of the tanks affect S in opposite directions. One possible explanation for this would be spacecraft symmetry characteristics. Similarly, the lateral inertia of the system ($dIyy\_body$ and $dIzz\_body$) drive the cost as well. However, this analysis shows the effects of each uncertainty in isolation. It is important to study the effects of the driving uncertainties combined together, with the eight redundant uncertainties removed.

Identifying the driving uncertainties is an important result in itself. It informs the designer of which uncertain parameters could be most problematic, and therefore gives an indication of where modelling effort is most useful. In addition, the driving un-

certainties are helpful in performance analysis, since the non driving uncertainties can be effectively eliminated from the system, potentially making it a more focussed and effective exercise. Therefore, further analysis is now done with only the driving uncertain parameters included. As with Figure 5.8, a DE based optimisation analysis, a CE based analysis, and a μ-based Robust Performance analysis are carried out.

The DE run is carried out with algorithmic parameters identical to the ones for the run with the full design uncertainty set, shown in Table 5.2. It results in a worst case performance of $J_{DE}^{S*} = 5.6382dB$, where the S superscript indicates the driving uncertainty set for S. This result is very close to the worst case performance found by DE for the full design uncertainty set is $J_{DE}^{*} = 5.6413dB$, with a relative percentage error of $\frac{5.6413-5.6382}{5.6413} * 100 = 0.054\%$. This small relative error indicates that the driving uncertainty set is found correctly, and that the non-driving uncertainties collectively account only for 0.054% of the performance function. The relative error may be reported as a margin of error when only the driving uncertainty result is presented. A plot showing the evolution of the maximum and mean performance values at every iteration of DE is shown in Figure 5.18. As previously done, note that the mean and maximum performance plots converge towards the end, and that there are no large jumps in performance after the 500th evaluation — both these factors suggest that the worst case found is close to the true worst case.

Next, a CE-based analysis is performed considering the driving uncertainty set. The algorithmic parameters chosen are kept identical to those in Figure 5.3. The values of γ considered for the analysis are also identical to the design uncertainty case. The probability profile of performance generated is shown in Figure 5.19. It is interesting to note that the worst case is again expected to lie between $\gamma = 5.6$ and $\gamma = 5.65$, as before with the CE analysis considering all design uncertainties. However, the probability

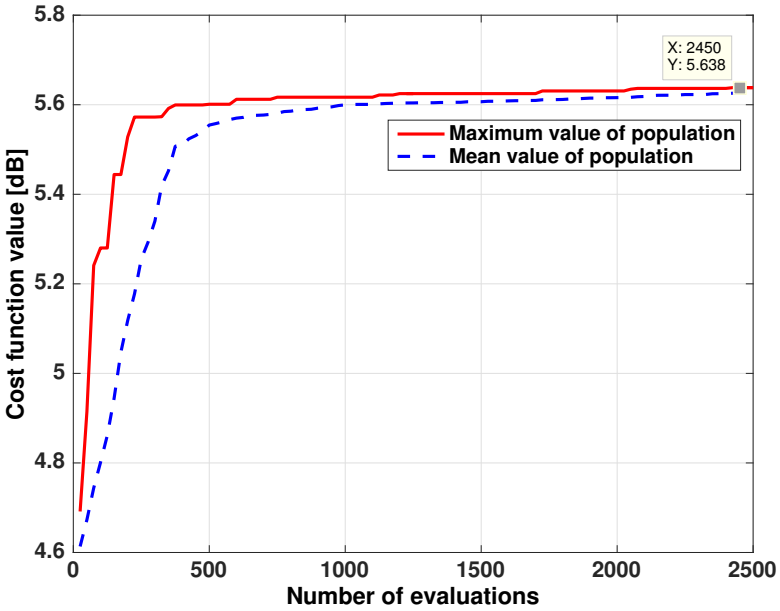


Figure 5.18: Results of the DE optimisation run considering only the driving uncertain parameters. Plot shows the evolution of performance values as iterations progress.

associated with $\gamma = 5.6$ is 3.738×10^{-8}, higher than that in the design uncertainty case, implying that when only the driving uncertainties are involved, the probability of $S_v(S)$ exceeding $5.6dB$ is just 4 in 100 million (up from 1 in a billion with all uncertainties considered). The final mean and standard deviation values found by CE at different values of γ are shown in Figure 5.20.

A μ-analysis run is now carried out. The slow algorithm is implemented, with the gain based method in `robustperf`. The upper and lower bounds computed are shown in Figure 5.21. The lower and upper bounds of the robust performance margin are 1.0069, and 1.00835 respectively. These bounds are fairly tight, and since the lower bound is above 1, robust performance is guaranteed. The worst case performance is $5.63dB$ at 0.125 rad/s, which is identical to that obtained with the full design uncertainty set.

Taking together the results of CE, DE, and μ-analysis , it is concluded that both de-

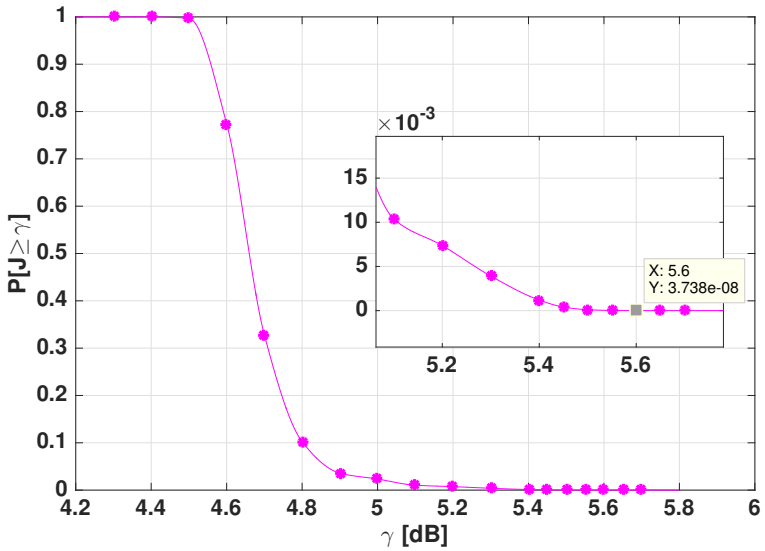


Figure 5.19: Probability profile of performance generated by CE considering only the driving uncertain parameters

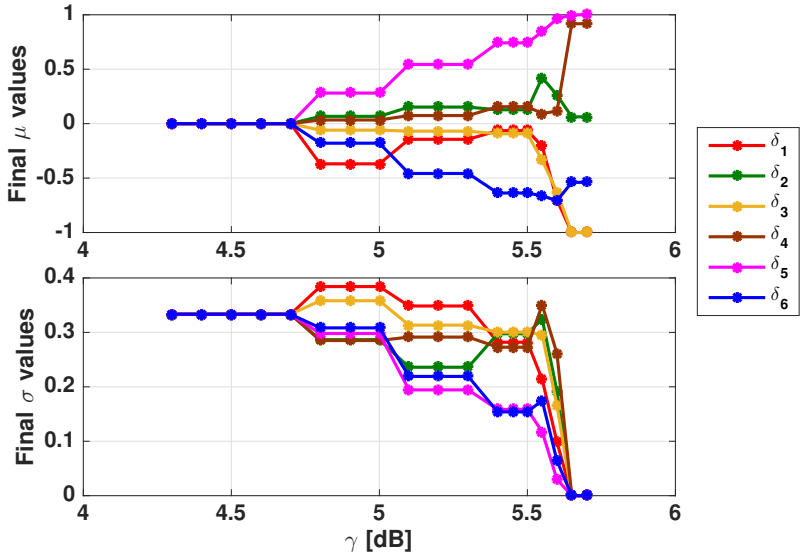


Figure 5.20: Final mean and standard deviation values of the driving uncertain parameters vs γ

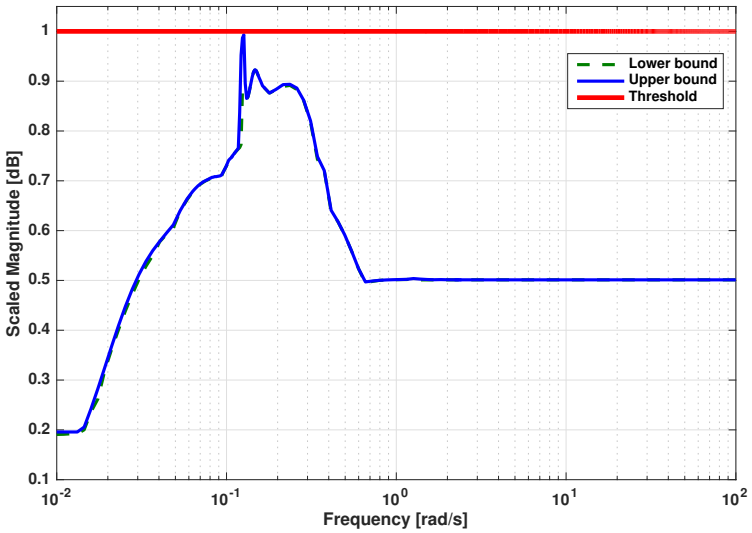


Figure 5.21: Upper and lower bounds computed by μ-analysis considering only the driving uncertainty set

terministic and probabilistic robust performance are guaranteed (since the worst case performance J is less than $6dB$ for all uncertain parameter values). A comparison of the worst case values found by CE, DE and μ-analysis for the driving uncertainties is shown in Figure 5.22. The three methods are comparable in terms of uncertain parameter directionality, implying that all three methods have identified the same parameter regions that lead to poor performance. The three methods have different internal mechanics, arrive at the same result, yet provide different insights into the performance of S. Hence these methods are complementary to each other.

The CE analysis affords an interesting insight into the parameter space and the performance function's behaviour in it. The final IS samples and costs are scattered, with two parameters taken at a time, as shown in Figure 5.23. The performance values are mapped to the colours, with dark red representing the highest values. This creates a heat plot-type representation of the parameter space, reminiscent of the safe and unsafe

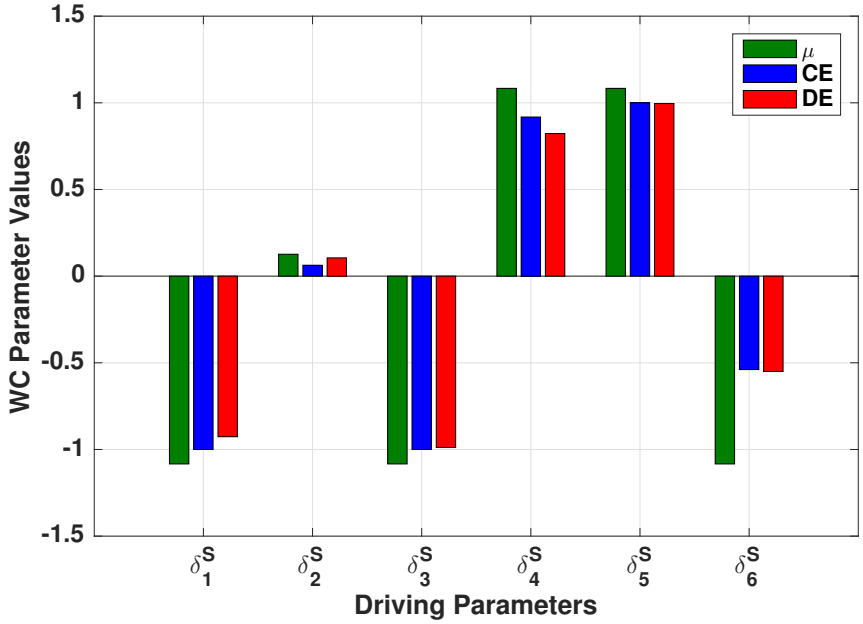


Figure 5.22: Comparison of worst case parameter values found by DE, CE and Robust Performance run using μ-analysis

region concepts [18]. The 3-dimensional plots at the top show the performance value with respect to two uncertain parameters. Beneath each three dimensional plot are the $x-z$ and $y-z$ views showing how each parameter impacts the performance value. Since the colours maps the performance values, all of the information retained in the three dimensional plots are retained, while affording a clear view of problematic parameter regions. The plots are particularly striking in the case of slosh stiffness parameters δ_5^S and δ_6^S. As their worst case regions are approached, there exists a high density of performances at and above $5.6dB$.

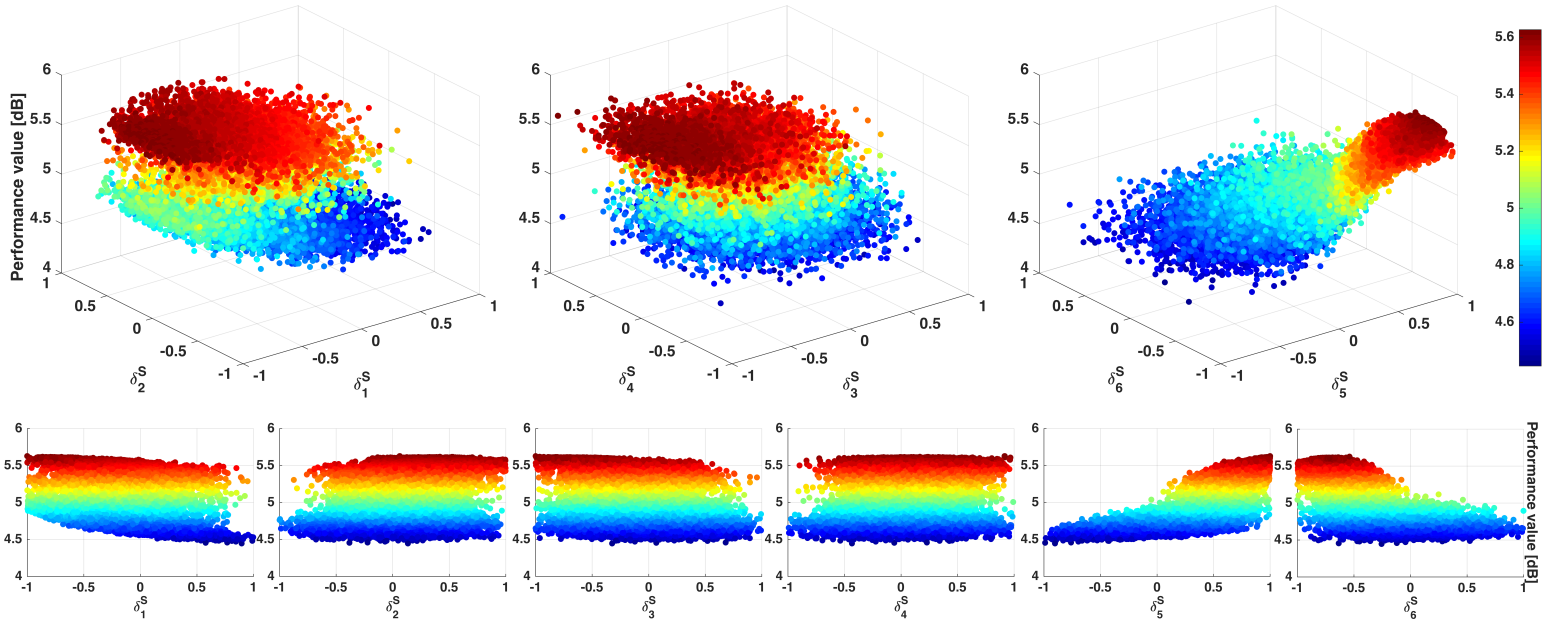
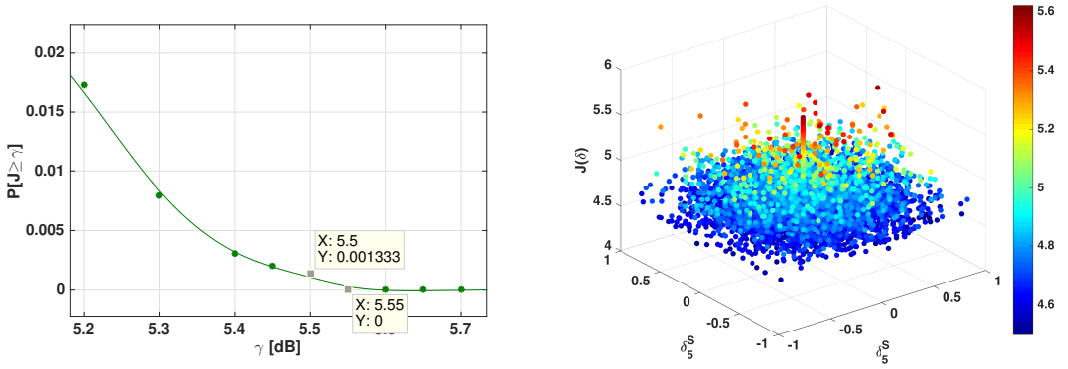


Figure 5.23: Heat plots showing performance function values in various uncertain parameter regions



(a) *Tail of the PPoP curve generated by MC with* $N_{cmc} = N + N_1$ *at each* γ.

(b) *Scatter plot from Monte Carlo data showing* δ_5^S *and* δ_6^S *samples versus performance*

Figure 5.24: Results of a classical Monte Carlo run, with $N_{cmc} = N + N_1$ samples.

The CE-based analysis results so far are compared with the results of a Monte Carlo run with $N_{cmc} = N + N_1$ evaluations at each value of γ. Since the true benefits of CE lie in the tail performances (the so-called critical region of the PPoP), Figure 5.24a shows the tail of the probabilities computed by MC, and 5.24b shows a scatter heat plot, generated in a similar way as 5.23. Clearly, the number of evaluations is not sufficient to create a clear idea of performance. In addition, the tail of the probability profile converges to zero at just $5.55dB$. Therefore, MC is far less capable at arriving at extreme performances than CE.

Apart from this preliminary comparison in terms of computational efficiency, it is noted that the formal comparison of MC and CE w.r.t. computational complexity and accuracy is out of the scope of this thesis. However, the literature on the subject is rich in both theoretical and practical details [66][94][92].

Having analysed the sensitivity function of the linearised autonomous rendezvous controller, and gained significant insight into its performance, the interest is in doing the

same for the complementary sensitivity function T. Before proceeding, an integrated analysis template for performance analysis of linearised systems is explored in the following section.

5.3.2 An Integrated Analysis Template

A theme emerges from the the analysis conducted, and the results obtained so far. The sequence of analyses carried out may be summarised in Figure 5.25, which is proposed as an integrated analysis template. By following such a template for performance analysis of a linearised system, the following values may be computed:

- the worst case performance,
- the probability associated with the worst case performance,
- the probability profile of performance (showing whether probabilistic robust performance is satisfied),
- the robust performance margins, and
- the driving uncertain parameters.

Depending on the need, the entire template may be followed, or only a part of it. For instance, it may be sufficient for some control systems to compute the worst case performance, the associated probability, and to identify the driving uncertainties. This may be considered for when the analysis must inform further design refinements. In another instance, it may be appropriate to start with a sensitivity analysis and arrive at the driving parameters first, and omit the entire first half of the analysis. However, this may result in not knowing with certainty the error in reducing the uncertainty set. In yet another case, the analytical $\boldsymbol{\mu}$-analysis may be omitted throughout the analysis, e.g., for industry projects whose requirements rely on statistical guarantees rather than

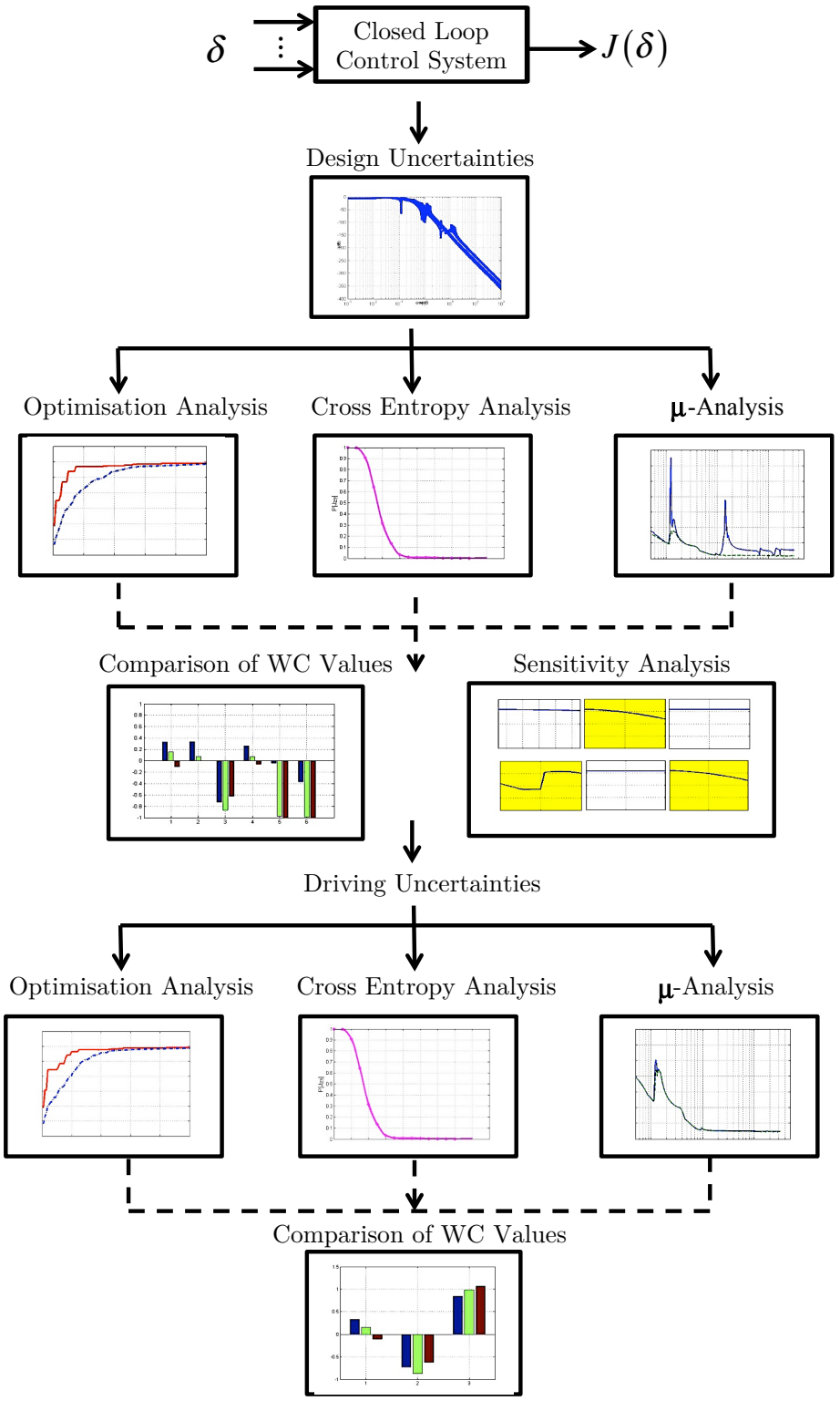
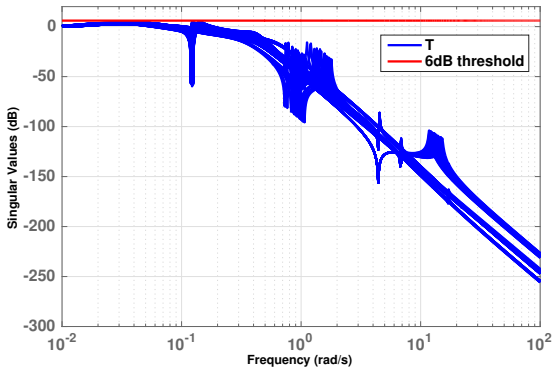
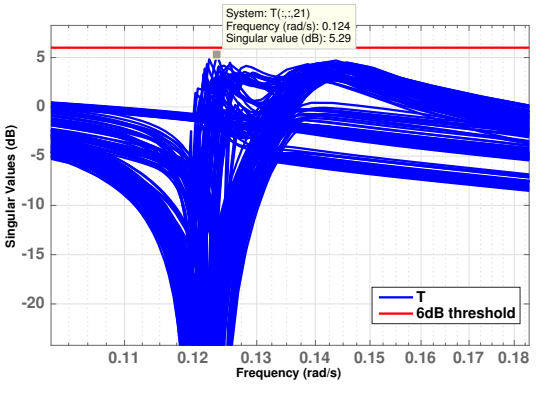


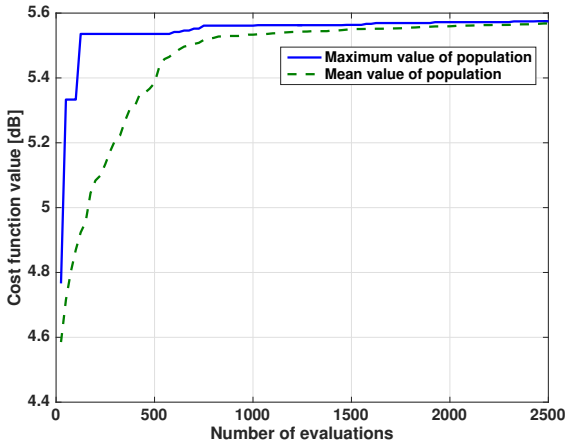
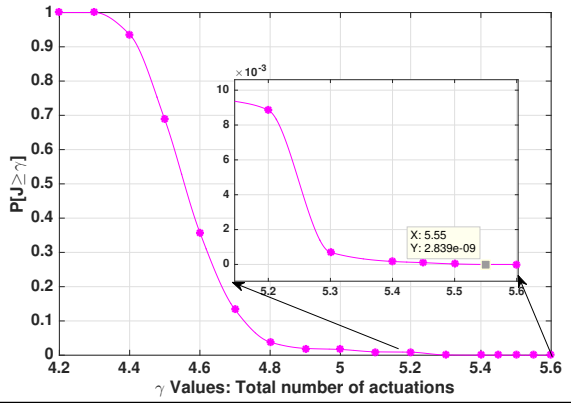
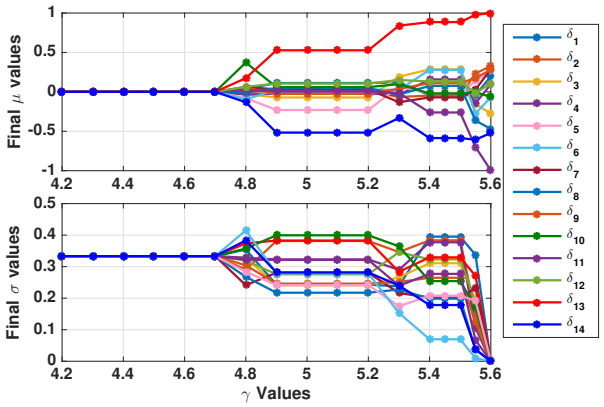
Figure 5.25: Integrated analysis template for performance analysis of linearised spacecraft systems

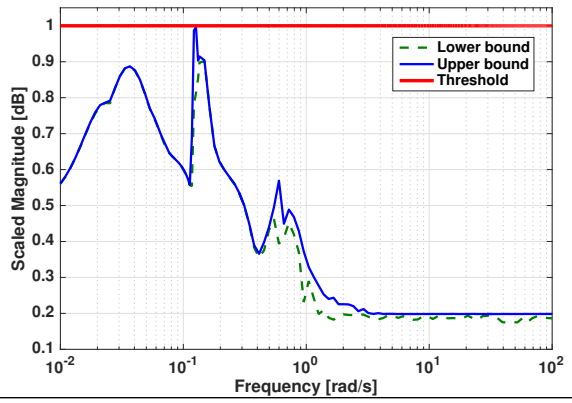
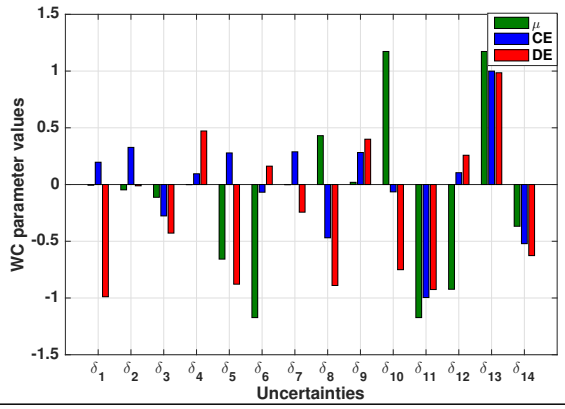
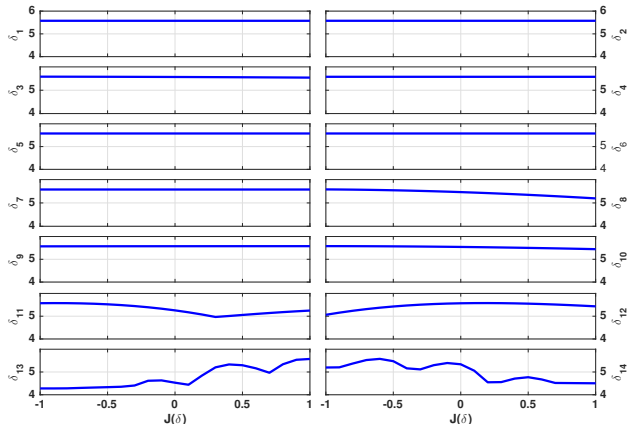
robust performance margins. The following section illustrates the use of the entire template for analysing the system w.r.t the second performance criterion.

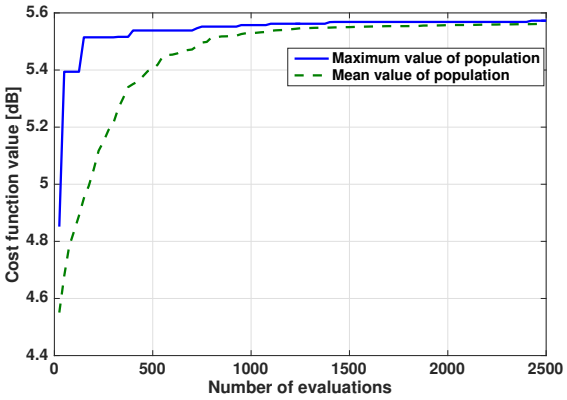
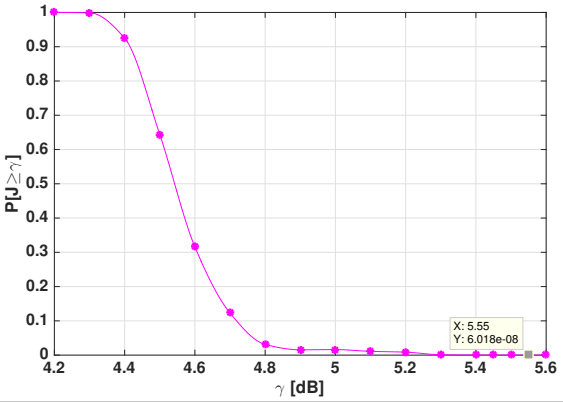
5.3.3 Complementary Sensitivity Function

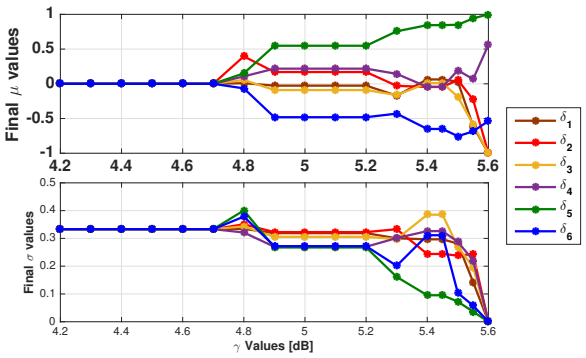
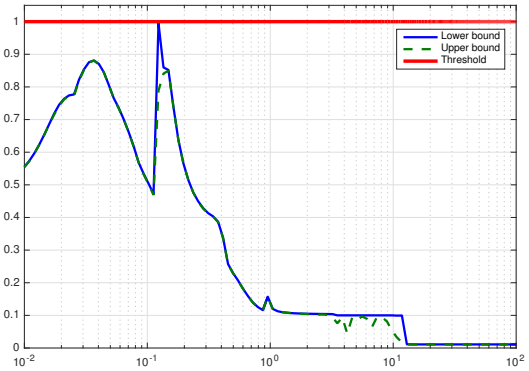
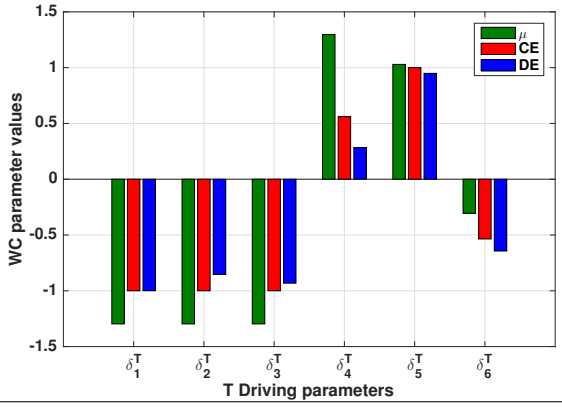
The template in Figure 5.25 is now followed sequentially for the Complementary Sensitivity Function T. The cost function considered here is the maximum singular value of the complementary sensitivity function T, which is desired to be less than 6dB.

Technique Settings	Remark	Plot
Singular values		
Singular values	$5.2294dB$ at $0.1236\ rad/s$	

DE iterations=100, pop = 25, crossover = 0.8, stepsize = 0.8	$5.5750dB$	
CE N=500, N_1=1000, ρ=0.1, σ^*=10^{-6}, t_{max}=30	$5.5641dB$	
CE μ_T **and** σ_T		

μ-analysis	UB=1.1728, LB=1.0043, freq=0.126rad/s, WC=5.5755dB	
Uncertainty Comparison		
Sensitivity Analysis		

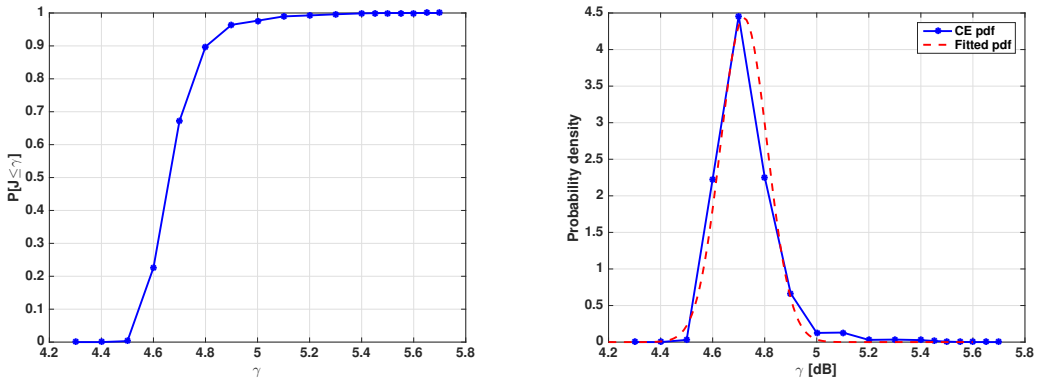
<table>
<tr>
<td>Driving Uncertainties</td>
<td></td>
<td>
<table>
<tr><th>Driving Uncertainties</th><th>Percentage Contribution to Cost</th><th>Uncertainty Name</th></tr>
<tr><td>$\delta_1^T = \delta_8$</td><td>7.4211</td><td>$dIzz\_body$</td></tr>
<tr><td>$\delta_2^T = \delta_{10}$</td><td>2.4560</td><td>$dmass$</td></tr>
<tr><td>$\delta_3^T = \delta_{11}$</td><td>12.2615</td><td>$dslosh\_mass\_sm1$</td></tr>
<tr><td>$\delta_4^T = \delta_{12}$</td><td>10.1675</td><td>$dslosh\_mass\_sm2$</td></tr>
<tr><td>$\delta_5^T = \delta_{13}$</td><td>30.2551</td><td>$dslosh\_stiffness\_sm1$</td></tr>
<tr><td>$\delta_6^T = \delta_{14}$</td><td>23.8092</td><td>$dslosh\_stiffness\_sm2$</td></tr>
</table>
</td>
</tr>
<tr>
<td>DE
iterations=100,
pop = 25,
crossover = 0.8,
stepsize = 0.8</td>
<td>$5.5727dB$</td>
<td> 
</td>
</tr>
<tr>
<td>CE N=500,
N_1=1000,
ρ=0.1,
σ^*=10^{-6},
t_{max}=30</td>
<td>5.5802dB</td>
<td> 
</td>
</tr>
</table>

CE μ_T and σ_T		
μ-analysis	UB=1.1355, LB=1.0046, freq=1.27rad/s, WC=5.5820dB	
Uncertainty comparison	Compatible directionality	

Summary		**Robust performance guaranteed** **WC Performance = 5.575dB** **P[5.55 $\geq \gamma >$ 5.6] $\approx 6 \times 10^{-8}$** **RP margins = 1.0046 and 1.1355** **Driving Uncertainties:** δ_8, δ_{10}, δ_{11}, δ_{12}, δ_{13}, δ_{14}

5.4 Discussions

In this section, potential interpretations of the probability profile of performance are discussed. Specifically, it briefly explores the use of the probability profile of performance as a tool for spacecraft control systems design and analysis. The first step is to take the complement of this curve, i.e., $1 - P[J(\delta) \geq \gamma]\ \forall\gamma$. This gives the familiar cumulative distribution function (c.d.f., or simply cdf) of the performance, and is shown by the blue curve in Figure 5.26b.



(a) *Sensitivity function cumulative density function from CE data*

(b) *Sensitivity function probability density function from CE data*

Figure 5.26: Estimated cumulative distribution function and probability density function of the iGNC sensitivity function.

The cdf is a plot of $P[J(\delta) \leq \gamma]$ vs γ. Another curve of interest is the probability

density function (pdf), which shows the relative likelihood of the performance taking a particular value [100]. This curve is denoted f_γ. Computing f_γ from CE data is done by integrating the cdf curve, and is shown in Figure 5.26a. A normal pdf is then fitted using the matlab function `normfit`, which is shown by the red dashed curve in Figure 5.26b. This fitted pdf has a mean of 4.72 and a standard deviation of 0.09. The exact probability associated with a probability may be found by computing the area under f_γ. For instance,

$$P\{\gamma_1 < \gamma \leq \gamma_2\} = \int_{\gamma_1}^{\gamma_2} f_\gamma d\gamma \tag{5.1}$$

The curve f_γ gives an indication of the mean performance levels. A measure often used in the space industry (and several others) is the 3-sigma measure, which looks at 3 standard deviations away from the mean value. Here, this is $4.72 - 3 \times 0.09 = 4.45$ and $4.72 + 3 \times 0.09 = 4.99$. This implies that when the uncertain parameters are sampled from the specified distributions (in this case the truncated normal distributions), a majority of the sensitivity function maximum singular values lie in the range [4.45,4.99] dB. This fact may be confirmed by looking at the probability profile of S in Figure 5.9. Having such a range of 3-σ performances could be useful in choosing the design criterion, and in relaxing it if there has been a violation of criteria. For instance, if the threshold was lower at $4.9dB$ in this study, it is clear that there would be a probability of violation of 0.031. If it is decided to relax the threshold, it would be recommended to be at $4.99dB$.

A caveat however, is that since the normal fit is not an exact match, the 3-sigma estimate is only a rough indicator. Indeed, the pdf curve shown in Figure 5.26 is fat-tailed, meaning that its tails are fatter than the tails of a normal distribution. Therefore, the

probabilities under the tails will in fact be greater than predicted. The curve fitting may be improved by having more γ points in the CE analysis, which comes at a higher computational cost.

So far, the performance analysis of the simplified rendezvous system comprising linearised models has been presented. The rest of this chapter presents a performance analysis of the designed controller on the high-fidelity FES model.

5.5 Performance Analysis Results for Nonlinear iGNC Model

The performance analysis of the real world model with the integrated GNC algorithms are detailed in this section. As before, the crucial TRDC mode is considered for this analysis. This controller is used in the final forced translation motion, and starts when the chaser is about 1000m away from the target. The controller is a full 6 degree of motion control, designed by accounting for the simplified linearised model described in the previous section. In this mode, robustness properties are critical to determine. Indeed, the success of the entire MSR mission depends on the final forced motion towards the OS, and any deviation outside bounds of expected performance can cause damage to the chaser, the OS, or both. Therefore, the performance analysis is critical to determine (a) whether the controller, designed according to the simplified model performs satisfactorily under nominal conditions, and (b) whether robust performance may be guaranteed in the presence of uncertainties in the real world model.

The performance criteria for the controller are determined by the consortium to be specified in terms of the relative positions and velocities at the time of capture (terminal time). The criteria are as follows:

- Lateral position error < 50mm
- Longitudinal velocity error < 10mm/s
- Lateral velocity < 2.8 mm/s

The cost function $J(n_e)$ at evaluation count n_e is therefore chosen such that it represents appropriately the deviation in terminal positions and velocities from those obtained in the nominal trajectory. $J(n_e)$ is expressed as follows:

$$J(n_e) = w_{J_1} \left\| \begin{bmatrix} v_y(nom) - v_y(n_e) \\ v_z(nom) - v_z(n_e) \end{bmatrix} \right\| + w_{J_2} \left\| v_x(nom) - v_x(n_e) \right\| + w_{J_3} \left\| \begin{bmatrix} x_y(nom) - x_y(n_e) \\ x_z(nom) - x_z(n_e) \end{bmatrix} \right\| + w_{J_4} \left\| \begin{bmatrix} \alpha_y(nom) - \alpha_y(n_e) \\ \alpha_z(nom) - \alpha_z(n_e) \end{bmatrix} \right\| \quad (5.2)$$

where x_y, x_z are the lateral positions of the chaser spacecraft, and v_x, v_y, v_z are its velocity values in the x-, y- and z- directions. α_y and α_z are the SC lateral angular displacements. The *nom* in parentheses indicates the nominal trajectory values, and its difference from the value at evaluation count n_e expresses the error. The second norm is utilised as a measure of the error. The weights w_{J_1} to w_{J_4} express the relative importance of each term in the cost function. In this study the weights are all considered to be equal to 0.25[1]. This cost function is a measure of the deviation from nominal in the position, velocities and angular positions. It effectively encapsulates all of the performance criteria.

The nominal values of the terminal trajectory are calculated with all uncertain parameter values set at 0. The nominal trajectory terminal values are listed below.

[1]It is good practice to ensure that the weights w_{J_1} to w_{J_4} sum to 1

- Longitudinal Velocity Accuracy is 9.4646×10^{-5} m/s,
- Lateral Position misalignment is 0.0053 m, and,
- Lateral Velocity Error is 2.2328×10^{-4} m/s

The allowed deviation due to effects of uncertainty are 0.1 m/s in the longitudinal velocity, 0.050 m in the lateral position, and 0.005 m/s in the lateral velocity error.

Having defined the cost function, the performance analysis of the TRDC controller is now presented. Ideally, a combination of the methods discussed in Chapter 2 could be applied. However, the real world iGNC model is complicated, making most of the methods unsuitable. The first constraint is the structure of the system — since it describes high-fidelity models, it is highly nonlinear, and includes stochastic factors to account for disturbances. Consequently, the model cannot be reformulated in the form of LFTs. This renders analytical robust control techniques unsuitable.

Further, due to the large number of uncertainties, techniques relying on likelihood ratios cannot be applied. The likelihood ratio rapidly degenerates for higher-order problems [13], since the likelihood ratios of each uncertainty is multiplied. This implies that methods such as CE cannot be directly applied in this instance.

Additionally, a single evaluation of the model takes about 7 minutes. This leads to very large computational demands, with even 100 evaluations — a considerably small number of evaluations, especially in the case of large-dimensional systems such as the one under study — taking nearly half a day to complete. Applying MC simulation methods in this scenario is therefore not a reliable option, since reasonable levels of confidence and accuracy demand large numbers of evaluations (e.g., a reasonable level of confidence and accuracy of 0.0 each requires nearly 1000 samples according to the

Chernoff bound, and nearly 10000 samples according to the Bernoulli bound. Such a run would take between 5 days to a month and a half to complete). Further, a high dimensional parameter space necessitates a large number of evaluations for classical Monte Carlo methods.

Due to the above constraints, it is clear that several of the available WCA techniques are ruled out. Optimisation algorithms remain a good choice for such challenging problems, since they intelligently search the parameter space for optima. For this reason, a very large number of samples is often not necessary, unlike with brute force methods such as the Monte Carlo or deterministic methods such as μ-analysis. A previous study describing the performance analysis of rendezvous and docking controller analysis [68] adopts a similar strategy.

Exeter's WCAT-II framework is used to carry out the analysis in the current study. An interface is designed between the GNCDE and WCAT-II for this purpose. This interface is tailored such that WCAT-II can access the simulation parametrisation and scattering, in the GNCDE environment, and then run the simulator. The GNCDE data files contain the scenario and system parametrization, which are used for the synthesis of uncertain state space (USS) models and the GNCDE template with the non-linear simulator.

The WCAT iterates with the GNCDE to carry out evaluations of the nonlinear simulator, setting the initialization of the parameters subject to the optimisation search. The process is iterative, with the chosen optimisation method evaluating the outputs of the simulation through a cost function defined for each campaign, and the optimisation search being tailored according to the resultant value. Such a V&V strategy combing analytical and optimisation-based methods is detailed in [11].

The rendezvous model is simulated from the Forced Motion (FM) to capture (CA) stages. A nominal circular target orbit is assumed, and it is assumed that the MAV orbit insertion has been successful. In this analysis, the optimisation algorithm selected is DE, whose working is described in Chapter 2, along with an added local optimisation search at the end. The uncertain parameters are scattered as per a uniform distribution, with the bounds at ± 1 for each normalised uncertain parameter.

The optimisation algorithm searches for the maximum value of the cost function (5.2), i.e, it maximises the error from the nominal value. This maximum constitutes the worst case cost, and the corresponding uncertain parameter vector is the combination of parameters that leads to the worst case. The simulation was carried out with the DE parameter values of crossover ratio 0.8, step-size 0.8, number of iterations as 20, and population size in each iteration as 10.

The optimisation run therefore spanned 200 evaluations, taking about 23 hours. Over each iteration, the cost function values of the population members are collected, and the highest cost values until that point are plotted as iterations progress — this plot is shown in Figure 5.27. The mean values of the population at each iteration is represented by the green dashed line. The maximum value and the mean values follow a similar trend. The plot reveals a steady rise in the cost until the 4^{th} iteration, at which the maximum was arrived at. In fact, the 38th function evaluation found this worst case value. The strength of randomised optimisation methods such as DE lie in exploring the search area using intelligent strategies and arriving at the optimum rapidly.

The worst case value is found to be 0.0685, which represents the maximum weighted

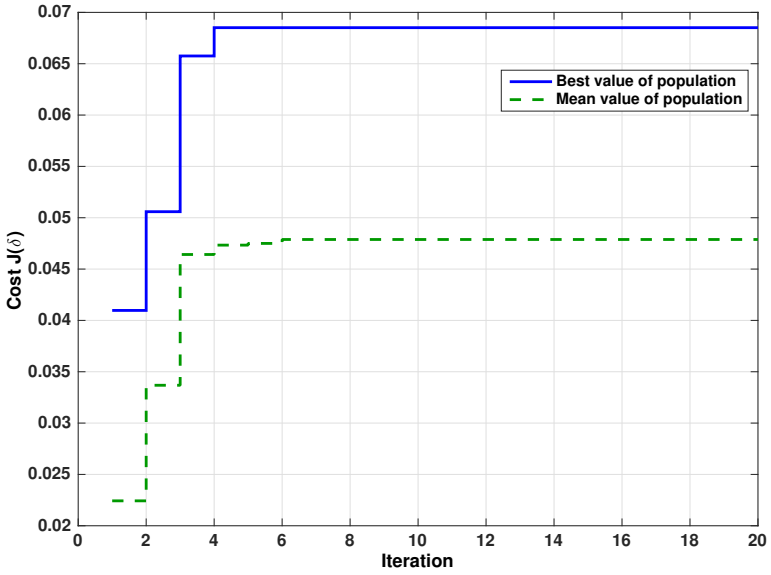
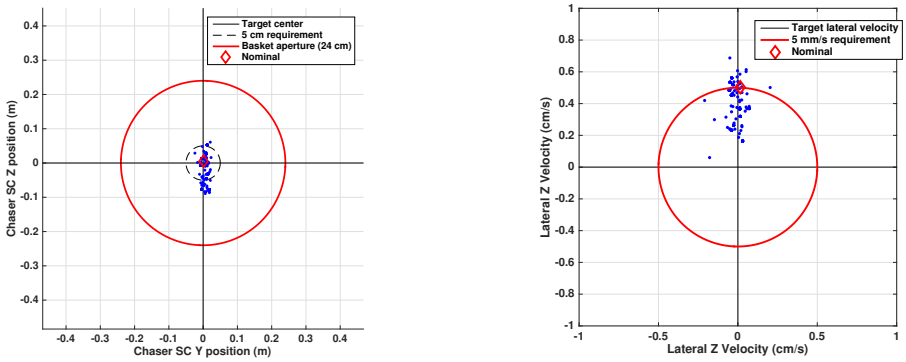


Figure 5.27: Best performance function values obtained by DE when implemented on the high-fidelity iGNC model

error from the nominal trajectory. The individual terms consisting of chaser position, attitude and velocity corresponding to this cost are as follows:

- Longitudinal position: 5.7209m
- Lateral position along y-: 0.01826m
- Lateral position along z- : 0.1113m
- Longitudinal velocity: -0.1072m
- Lateral velocity along y-: -0.0006m
- Lateral velocity along z-: 0.0024m
- Angular displacement along y-:-0.0061 deg
- Angular displacement along z-: -0.0232 deg



(a) Lateral position error

(b) Lateral velocity error at terminal time

Figure 5.28: Lateral position and velocity responses

It is observed that these performances remain within the required bounds. The lateral position capture performance is depicted in Figure 5.28a, while Figure 5.28b shows the lateral velocity performance. These figures show the values of chaser SC positions and velocities at the final time, across all the valuations. Observe that the lateral misalignments are within the bounds in most cases, except for a few violations of the bound. These violations, and the corresponding uncertain parameter values are reported to the consortium, and are taken as inputs for any further controller retuning. There appears to be a small bias towards the positive Z direction in the terminal relative velocity, and indeed the nominal value is at the bound. One possible explanation for this is that effects such as flexible modes and sloshing cause a vertical disturbance not sufficiently accounted for in the controller. In addition, the controller itself may allow for a vertical movement, since the nominal value also is biased vertically.

The longitudinal velocity values at the final time are shown in Figure 5.29. Both the nominal value and the perturbed trajectory values are well within the required bounds, implying that the controller behaves desirably with regard to the longitudinal direction.

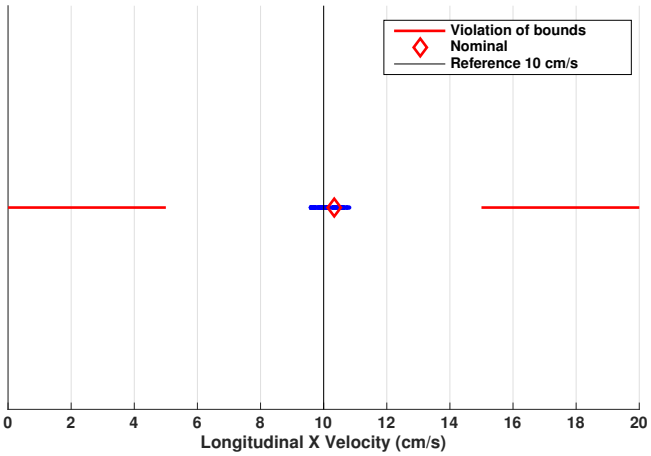


Figure 5.29: Longitudinal position misalignment

The position trajectory histories show that the entire trajectory in Figure 5.30 are close to the nominal trajectory (shown in the thick red line) across all the evaluations. The velocity histories of all the perturbed trajectories are shown in Figures 5.31. In both cases, some perturbed trajectories deviate significantly until about the 500th second, which are attributed to uncertainties, since the nominal trajectory (shown in thick red line) does not deviate greatly. However, from the 500th to 1000th second, the behaviour is close to nominal.

The incidence angle of the chaser SC w.r.t the target is important — a large angle of incidence can potentially damage the OS or the SC, particularly if the impact velocity is high. Figure 5.32 shows the incidence angle performance at the final time. The two-degree bound is violated several times. Interestingly, some of the cases have high incidence angles of upto 3.5 degrees. This effect does not register in the worst case value, however, possibly due to being outweighed in the cost function by favourable

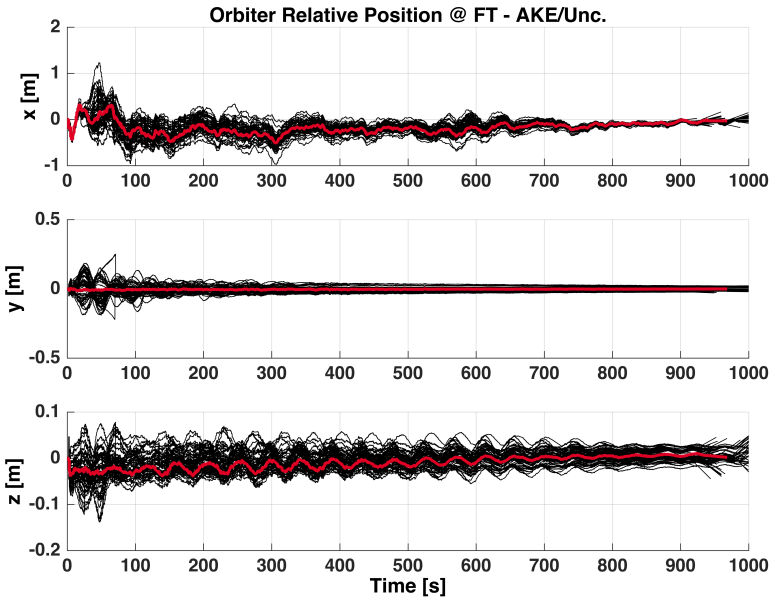


Figure 5.30: Position trajectory histories

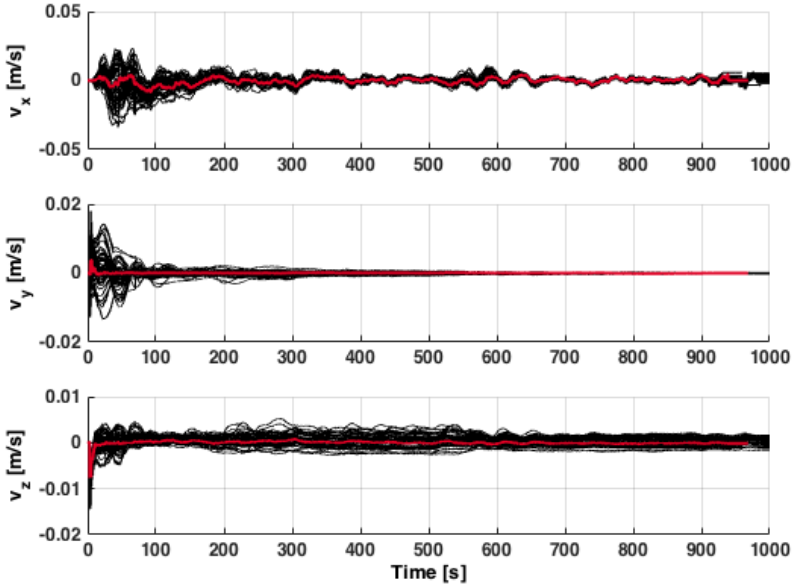


Figure 5.31: Velocity trajectory

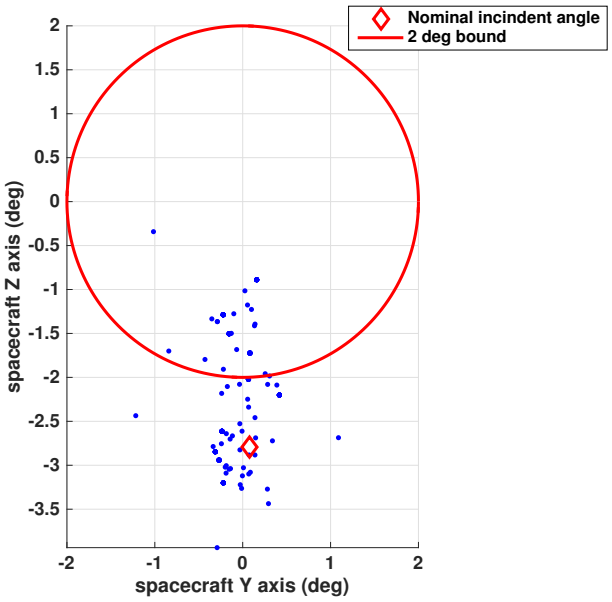


Figure 5.32: Lateral incidence angle performance

performance in the other effects.

Such effects may also be detected in the early parts of the attitude trajectory histories, shown in in Figure 5.33, and the angular velocity trajectory histories in 5.34. Note the large deviation from the nominal trajectories for the first 200 seconds in both plots. Such effects could be attributed to disturbance torques and rotational participation due to sloshing and flexible modes. The angular velocity trajectories (including nominal trajectories) continue to vary slightly until the end, while the attitude settles after about the 500^{th} second. Since the controller performance is only evaluated in terms of terminal performance, the initial variations are not in the scope of the current analysis. However, it is possible that larger deviations at the start of the trajectory carry over into terminal velocities due to not being damped sufficiently.

The performance analysis results presented so far have provided sufficient insight into

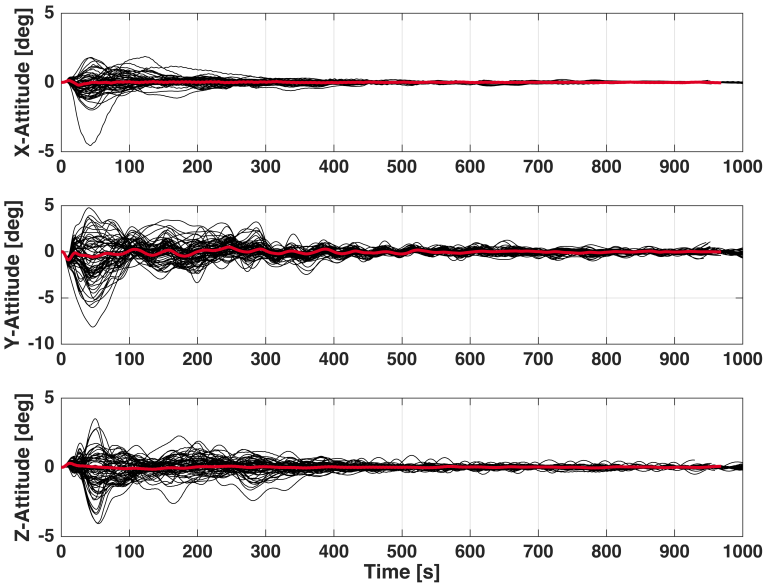


Figure 5.33: Lateral position trajectory

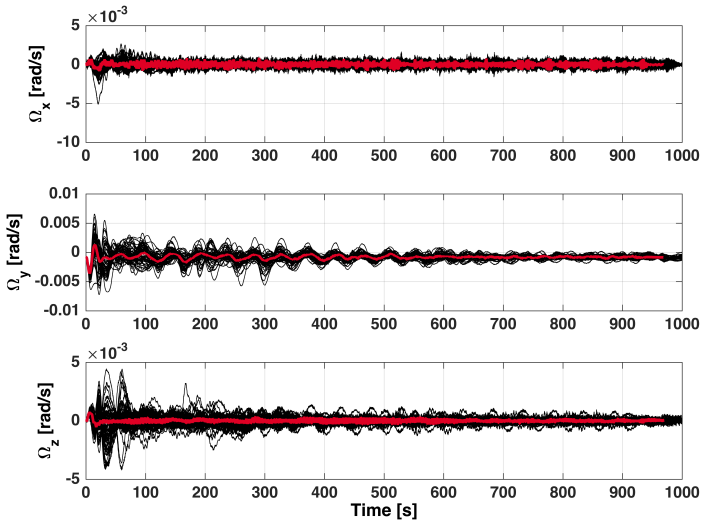


Figure 5.34: Angular velocity trajectory histories

the TRDC controller performance, especially problematic performance cases. These results demonstrated that the designed controller performed reasonably well, with a few violations of the requirements. A question arises as to whether the worst case value obtained may be improved upon. However, due to the heavy computational constraints, a more rigorous optimisation run is infeasible.

A strategy used to improve the worst case results is to include μ-analysis results conducted during the controller synthesis, introduced in previous ESA projects conducted at Exeter [11]. In the iGNC project, $\mu-$analysis is carried out as part of the controller synthesis, for the 14 design uncertain parameters accounted for in the process. During the subsequent optimisation-based worst case analysis, the nominal values of these parameters are fixed at the uncertain parameter values provided by the analysis. This way, the optimisation effectively searches in their neighbourhood.

The values of uncertain parameters obtained during the synthesis process are provided to Exeter by TASF. The computed worst case parameter values are in the second column in Table 5.9. The values outside of the parameter bounds [-1,1] are rounded down to ± 1. It is important to bear in mind that the worst case parameter values found by $\mu-$analysis are different in at least two major ways:

- The results depend on the linearisation points of a simplified system — for the dynamics, sensors and actuators, and the GNC.
- The analysis is carried out in the frequency domain, and so neglect time domain uncertainty effects such as initial conditions.

Nonetheless, the parameter values can be good indicators of the direction of worst case, along which a search can reveal worst cases more efficiently. Further, incorporating them as initial values in the optimisation-based analysis is certainly not detrimental

Table 5.9: μ-analysis values to enhance performance of optimisation-based WCA

Parameter	Analytical WC Value	Bound for DE initial value
$dBR\_sa\_mode\_y\_1$	+1.0641	+1
$dBT\_sa\_mode\_x\_1$	+1.0641	+1
$dIxx\_SA$	-1.0641	-1
$dIxx\_body$	-1.0641	-1
$dIyy\_SA$	1.0641	1
$dIyy\_body$	1.0641	1
$dIzz\_SA$	-1.0641	-1
$dIzz\_body$	-1.0641	-1
$dfreq\_sa\_1$	-1.0641	-1
$dmass$	-1.0641	-1
$dslosh\_mass\_sm1$	-1.0641	-1
$dslosh\_mass\_sm2$	1.0641	+1
$dslosh\_stiff\_sm1$	0.7180	0.7180
$dslosh\_stiff\_sm2$	-1.0641	-1

Table 5.10: Worst case costs and terminal position and velocities for purely optimisation based WCA and enhanced optimisation based WCA.

Simulation	**Cost**	y **[m]**	z **[m]**	v_x **[m/s]**	v_y **[m/s]**	v_z **[m/s]**	α_y **[deg]**	α_z **[deg]**
Optimisation	0.0685	0.01826	0.1113	-0.1072	-0.0006	0.0024	-0.0061	-0.0232
Enhanced	0.09916	-0.0018	-0.0957	-0.1005	0.0015	-0.0037	0.8624	2.0969

to the analysis, as the can move the values if there are more favourable values found during analysis.

The optimisation strategy, the DE settings used, the orbit considered and initial values are identical to the above case. The results obtained are in Table 5.10. It is observed that the enhanced scheme gives a worst case cost value of 0.09916, an improvement of 44.7% over the purely optimisation strategy. Note that even at this higher cost, all of the positions and velocities are within the required bounds. Only the incidence angle along the chaser z-axis marginally exceeds the 2 degree requirement. This term contributes to the high cost obtained.

It must be noted that the project partner GMV carried out an extensive Monte Carlo

campaign [119] consisting between 500 and 1000 evaluations with the uncertain parameters scattered as per Table 4.2. The worst case value found was 0.0854, lower than the one found via the enhanced scheme. This suggests that the enhanced analytical-optimisation strategy is very effective in finding worst cases. The Monte Carlo scheme is useful when computational constraints are not tight as in this problem.

The performance analysis results shown in this section suggest that the designed controller satisfies many of the performance criteria satisfactorily. Although a gentle vertical bias is discerned in the position and velocity, it is within the specified bounds and thus acceptable. The attitude performance showed the most number of violations, and must be investigated further.

5.6 Conclusions

The results of robust performance analysis of the crucial TRDC mode closed loop control system of iGNC is described in this chapter. The robust performance analysis of both the early design stage simplified rendezvous model, and the high fidelity FES model is carried out.

In the first part of the chapter, the designed controller is assessed for robust performance on the simplified rendezvous model. The performance objectives are expressed in terms of the crucial sensitivity and complementary sensitivity functions. The analysis is carried out using DE, CE and $\boldsymbol{\mu}$-analysis. The study finds that the designed controller achieves robust performance in the presence of modelled uncertainties, w.r.t both the sensitivity and complementary sensitivity functions. In addition, it is also found that results computed by CE tie in with technologically mature tools such as $\boldsymbol{\mu}$-analysis and DE, confirming its accuracy. Further, it is found that CE is capable of

providing rich insight into controller performance, in terms of probabilities associated with performance as well as its dependence on the parameter space. The important problem of quantifying conservatism may be addressed by applying CE. CE-based analysis is, therefore, a good choice for probability based performance analysis of spacecraft controllers.

The CE-based method is not without limitations, however. It has been observed that for problems greater than 30 parameters, the likelihood ratio degenerates, which is also termed the "curse of dimensionality" [92]. In addition, the run can take significantly longer than a DE analysis, depending on the number of performance levels considered and the standard deviation threshold. However, it may be argued that this increase in computation effort is justified by the extra insight the technique provides.

A performance analysis of the nonlinear model in the crucial TRDC mode with the closed loop controller designed by the project partners is carried out using optimisation based methods, and it is found that the controller performance is mostly satisfactory. The worst case performance is within acceptable bounds, and the effect of uncertainty does not lead to troublesome behaviour. An enhanced optimisation analysis with μ-analysis worst case parameters included as initial values is performed, and is observed to be effective in finding higher values of the worst case. Further, both strategies, i.e., the optimisation-only and enhanced analytical-optimisation schemes arrive at worst case values much quicker than Monte Carlo analysis.

However, it is worth emphasising that the model has a large number of uncertainties (over 200) in the system, and each function evaluation taking several minutes for computation. Due to this, the performance analysis is complicated by computational constraints. For this reason, neither analytical methods such as μ-analysis, nor prob-

abilistic methods or Monte Carlo can be reliably applied. The simplified rendezvous model introduced in this chapter, however, is suitable for analysis using various methods.

In the first part of this study, a simplified system is deliberately chosen to establish CE as an effective technique for analysis. In the following chapter, contributions made to two space industry projects using cross-entropy based analysis are described.

Chapter 6

CE Applications in Launch Vehicle Performance Analysis

Previous chapters have introduced the CE-based simulation method to estimate the probability associated with rare events. The method has then been implemented on the linearised autonomous rendezvous controller to generate its probability profile of performance and identify the probability associated with the worst case. In this chapter, CE is implemented in two industrial launcher models. The first model is an attitude controller in the long coasting phase of a launcher upper stage. This analysis, carried out in the early stage of controller synthesis, investigates the performance of a Dynamic Inversion (DI) controller designed to maintain the attitude during spinning motion necessary in long-coasting upper stages. This study, although carried out in a nearly linear environment, is nonetheless interesting for two reasons. Firstly, it accounts for launcher payload configurations as configuration parameters in the analysis, along with the modelling uncertainties typically included in robust performance analysis. In addition, the performance function - the total number of thruster actuations commanded by the controller - is restricted to integer values. Both these factors are important for launch vehicle analysis, in general, and a $V\&V$ method that can account

for these is useful [120].

The second model is that of Vega, a commercial launch vehicle developed by the European Launch Vehicle (ELV). The dynamics of Vega are emulated by a high-fidelity non-linear simulator. In this study, the objective is to verify that, in the first 60 seconds of flight, the aerodynamics do not exceed safe limits. Two investigations are undertaken: first, a small subset of four critical parameters is considered, in order to compute the probability associated with various γ levels, from satisfactory to degraded performance. A real wind model is included in the assessment. The second investigation studies the effects of wind on the structural loads experienced by the launcher first stage. This part of the study compares effects of four real wind models and accounts for a larger uncertainty set with 28 parameters.

The investigations have the following objectives:

- to assess whether the controller performances of the two industrial systems meet the desired performance criteria,
- to increase confidence in the CE method for robust performance analysis in industrial spacecraft models, and
- to standardise the CE algorithm by integrating it into WCAT-II and make it suitable for future applications in aerospace industrial $V\&V$.

Note that the results presented in this chapter are not comprehensive, and require deeper analysis to gain insightful understanding of both systems' performances. The two studies do, however, improve the technological maturity of CE (an important requirement in safety-critical applications). The investigations aim to serve as further evidence for the efficacy of CE.

The chapter begins with a brief description of the long coasting launcher upper stage dynamics, and the designed controller. The parameters included in the study, the performance function considered, and the Simulink model employed are then described. The results of optimisation- and CE-based robust performance analysis are subsequently presented. The implications of performance-bound violation are then discussed, with the aid of probability information found by CE. The next part of the chapter introduces the Vega launcher model and states the aims of its performance analysis. The performance criterion and the uncertain parameter set is then described. The results of the analysis are then presented.

6.1 Launcher barbecue mode attitude control system model

Future geostationary launchers (such as the Ariane-6 currently under development) are expected to fly payloads directly into the geostationary orbit. Due to this, a long coasting phase lasting up to 5 hours is included, in which the upper stage drifts with its payloads towards the point on the orbit where separation takes place. A long drift phase exposes the upper stage to dangers of thermal stresses, since one side of the launch vehicle's body faces the sun and the other faces away. To prevent such thermal stresses, the stage slowly rotates around its roll axis with a spin rate of 1-5 deg/sec. An attitude controller mode known as the barbecue mode is invoked [121]. This section presents the robust performance assessment of a conceptual control design for the barbecue mode.

The study presented in this section was carried out during an on-site collaboration with Airbus Defence and Space, Bremen in the frame of the project "Upper Stage Attitude Control Design Framework — USACDF"[121] contracted by ESA. An objective of the project is to incorporate a high level of mission autonomy[120]. Towards this end, the

controller development process relies on auto-coding and auto-reporting, and is done in a unified, multidisciplinary environment. An advantage of high mission autonomy is that human involvement is minimised, therefore reducing development time, costs and manual errors. Consequently, $V\&V$ is all the more crucial in this context, since undesirable effects may be introduced by auto-coding, which need to be detected before flight. Note that the author of this thesis participated in one of the activities in the project, and the CE Matlab code remains with the team at Airbus Defence and Space, where it is currently being used in the rest of the project activities.

The launch vehicle upper stage controller is responsible for orbital injection, and for the first stage separation. The control is thruster actuated. Depending on the types of payloads an operational upper stage can have a wide range of allowable configurations in terms of MCI parameters and spin rates, all of which change after each separation. The implemented control algorithm must ideally adapt to change in payload configurations in real-time, because it is desirable to avoid retuning controller gains prior to each launch. Therefore, it is important to validate that the designed control law fulfils the performance requirement for all configurations. If challenging configurations exist, it is advantageous to identify them early in the design stage, so as to direct future control design and tuning iterations. In addition to these configuration parameters, there exist the classical uncertain parameters due to modelling inaccuracy. In this study, uncertain parameters whose nominal values are themselves configuration parameters are known as dependant parameters. These classes of parameters are explained in Section 6.1.2.

6.1.1 Description of the Model

A spinning upper stage, when acted upon by disturbance torques, oscillates in accordance with dynamics known as precession and nutation. Precession occurs when the

spin axis deviates from the roll axis of the upper stage, due to which the upper stage may exhibit a conic spinning behaviour. Nutation is the swaying oscillation that occurs along the circle described by the conic movement. These two effects are illustrated in Figure 6.1. If the disturbance torques are large, such motions can lead to instability. It is desired to avoid these effects, and to maintain the attitude angles at a pre-defined reference.

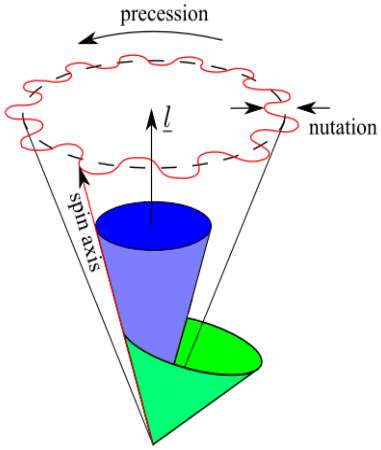


Figure 6.1: Illustration of precession and nutation of a launcher upper stage. Source: Airbus Defence and Space.

A designed barbecue mode controller should meet the following objectives:

- the attitude spin rate requirement should be met in the presence of disturbance torques,
- modelling uncertainties should be accounted for, such as the uncertainty in knowledge of lateral moments of inertia,
- the total number of thruster actuations must be maintained below a desired bound, and,
- these requirements should be satisfied for all possible payload configurations.

The model of the spinning upper stage dynamics in the barbecue mode can be found in standard texts on spacecraft dynamics such as [122], and is written as:

$$\begin{bmatrix} \dot{\theta}_y \\ \dot{\theta}_z \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} 0 & \omega_s & 1 & 0 \\ -\omega_s & 0 & 0 & 1 \\ 0 & 0 & 0 & \omega_n \\ 0 & 0 & -\omega_n & 0 \end{bmatrix} \begin{bmatrix} \theta_y \\ \theta_z \\ \omega_y \\ \omega_z \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ I_{2,2}^{-1} & 0 \\ 0 & I_{3,3}^{-1} \end{bmatrix} \begin{bmatrix} N_2 \\ N_3 \end{bmatrix}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_y \\ \theta_z \\ \omega_y \\ \omega_z \end{bmatrix} \tag{6.1}$$

where θ_y is projection of the body y axis onto the commanded spin axis direction and θ_z the corresponding projection of the body z. θ_y and θ_z define the precession cone, about which the spacecraft performs a nutating motion. In ideal conditions, θ_y, θ_z=0, and the precession cone is non-existent. Observe that when the spin rate $\omega_s = 0$, the MIMO system reduces to two decoupled double integrators. The pointing requirement is given by:

$$\left(\theta_y^2 + \theta_z^2\right)^{\frac{1}{2}} < \alpha_r \tag{6.2}$$

where α_r is usually in the range of $10 - 20$ deg. ω_y and ω_z are the lateral angular rates in the upper stage body axis. N_2 and N_3 are the lateral torque inputs along y- and z-axes respectively. ω_s is the x-axis spin rate, and ω_n is the nutation frequency defined as:

$$\omega_n = \left(1 - \frac{I_{xx0}}{\frac{I_{yy0}+I_{zz0}}{2}}\right)\omega_s \tag{6.3}$$

I_{xx0} is the nominal longitudinal moment of inertia, while I_{yy0} and I_{zz0} are the nominal lateral moments of inertia of the upper stage. The parameter I_{xx0} is payload dependent, and can range from about 0.2 to 0.8. The nominal value I_{xx0} is considered to be a configuration parameter. In addition, I_{xx} is considered a modelling uncertainty to account for measurement inaccuracies. I_{xx0} along with the spin rate ω_s and the lateral inertias I_{yy} and I_{zz}, are different at each launch, and all of these change after a separation event.

This section now describes the barbecue mode attitude controller. The control law is implemented based on Dynamic Inversion (DI) [121]. Details of DI are not relevant to this analysis, but the idea is that some transformation of the dynamics leads to an analytical expression for a state feedback law. The state feedback controller gain is derived in [121], and is given by:

$$K_G = \begin{bmatrix} K_{G_1} - \omega_s^2 & K_{G_2}\omega_s & K_{G_2} & \omega_s + \omega_n \\ -K_{G_2}\omega_s & K_{G_1} - \omega_s^2 & -(\omega_s + \omega_n) & K_{G_2} \end{bmatrix} \tag{6.4}$$

where K_{G_1} and K_{G_2} are the controller gains, to be specified and tuned by the control designer. The rationale for using DI is that it produces an analytical expression for the control, instead of intensive techniques such as linear quadratic controllers[1]. Further, it is assumed that the inertial measurement units (IMU) of launchers are usually quite good for the phase under study [121]. Therefore the state vector may be assumed to be accessible. However, DI is known to have rather poor robustness properties[123], and so the performance must be carefully assessed.

The controller is connected in feedback with the dynamics in Equation (6.1), and the resulting time response is shown in Figure 6.2. Notice that the open loop response has

[1] In these methods, the solution of the Ricatti equation is to be calculated offline and stored prior to each launch

an increasing oscillatory behaviour, which if left uncontrolled decays to infinity. As seen in Figure 6.2, closing the loop with the designed controller regulates the nominal plant's attitude at 0 degrees.

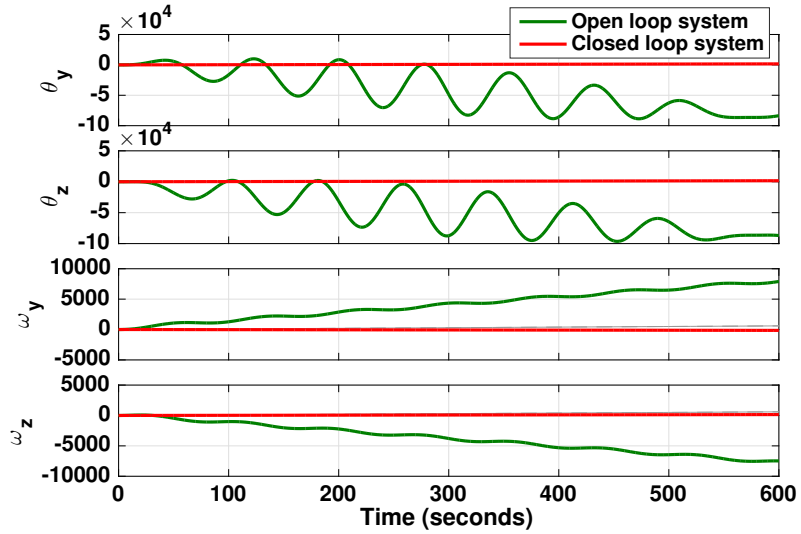


Figure 6.2: Time domain responses of the open loop and closed loop systems.

The control law K_G in Equation (6.4) can be used for a spinning payload release (which can be as high as $\omega_s = 30 deg/sec$ in certain cases). However, such an unmodified control law based on Equation (6.4) would command a very high number of thruster actuations over a couple of hours. This could be reduced by selecting a very low controller bandwidth, but there exists a trade-off between the control bandwidth and speed of response. Instead, an actuator threshold is introduced as follows:

$$N_{cmd} = \left\{ \begin{array}{rcll} 0 & : & if & N_G \leq N_{thr} \\ N_G & : & if & N_G \geq N_{thr} \end{array} \right\}, \tag{6.5}$$

where N_G is the torque calculated from Equation (6.4), and N_{thr} is the threshold above which the thruster is turned on. N_{cmd} is the finally commanded torque. Such a thresh-

old is called the thruster minimum impulse bit (MiB), and models the smallest control torque that can be commanded to the thruster, and avoids commanding for small attitude errors. This essentially introduces a dead zone non-linearity into the system. Further, N_{thr} is considered a design parameter for the control law.

Figure A.1 in the Appendix shows the Simulink model of the closed loop attitude control system of the long coasting upper stage. So far, the plant and controller described have been treated as continuous. However, to account for the sampling rate of sensors, a zero-order hold is introduced in the model. This renders the dynamics behaviour discrete. In addition, the thruster MiB is included separately in N_2 and N_3 control computations.

Taking these effects into account, the computed control input N_2 is as shown in Figure 6.3. Observe the discrete behaviour of the control inputs. These inputs are in the form of short torque impulses, which are delivered by the thrusters via on/off responses. Each distinct spike in the control is delivered by a single thruster actuation. There exist several instances of zero control, reflecting the actuation threshold. The magnitude of the control is between 0.05 and $0.1Nm$ in all cases. Although the N_3 plot is not shown, the behaviour is similar. The total number of actuations of N_2 and N_3 together must be within safe bounds, to ensure that thrusters are undamaged. Indeed, the performance criterion in this study is the total number of actuations of the thrusters delivering the N_2 and N_3 commands.

Before proceeding to investigate whether the controller meets its objectives, the parameters included in the model and the performance criterion need to be specified. The following section describes the three classes of parameters.

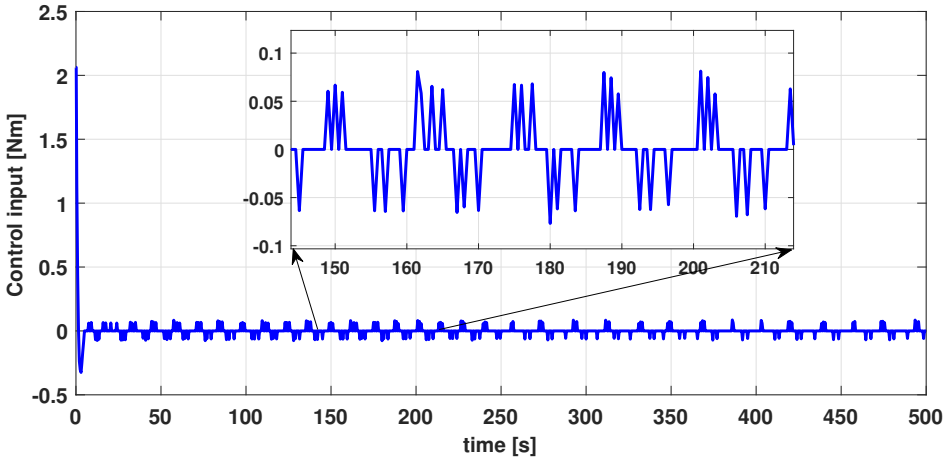


Figure 6.3: Control inputs computed by the dynamic inversion-based barbecue mode attitude controller.

6.1.2 Uncertain, configuration and dependent parameters

In order to account for the different types of parameters in this model, three classes of parameters are distinguished: configuration parameters, dependant parameters and uncertain parameters. Several of these parameters have already been mentioned in Section 6.1.1. Configuration parameters comprise the variations caused by payloads, which vary at every launch. Considering such parameters in $V\&V$ enables the control design team to anticipate and design for challenging configurations. The nominal value of the payload inertia, i.e., I_{xx0} is an important configuration parameter, as its value directly affects the nutation behaviour, as observed in Equation (6.3). The second configuration parameter is x_{com0}, the longitudinal distance from the centre of mass to the thruster. This distance is the lever arm which, when multiplied by the thruster force, delivers the control torques to the upper stage. x_{com0} is payload dependent because the centre of mass shifts for different payload sizes. The third payload dependant configuration parameter is the moment of inertia of the thruster mount I_{mib}. The MiB N_{thr} is inversely proportional to I_{mib}.

In addition to payload-dependant variations, there also exists an uncertainty in measuring I_{xx} and x_{com}. Since their nominal values are configuration parameters, these two uncertain parameters are called dependant parameters. In addition to the configuration and dependant parameters, there exist the familiar modelling uncertain parameters, i.e. parameters whose values are not known with precision. The lateral inertia I_{yy} and I_{zz} are uncertain, as is the spin rate ω_s. The misalignment of the thrusters can result in uncertain cross coupled torques, expressed as byz_{coup} and bzy_{coup}. The actuation to deliver the input torques N_2 and N_3 is generated by a thruster force F_{max}, which is considered to be an uncertain parameter.

The parameters described above are listed in Table 6.1, along with their nominal values, ranges and units. The values are established from system expertise. The configuration parameters depend on the payload, and are typically known with precision on launch day. Incorporating these parameters into the analysis ensures that all payload configurations are accounted for systematically, and so that challenging configurations may be identified.

Table 6.1: Uncertain, configuration and dependent parameters accounted for in performance analysis of upper stage barbecue mode.

Parameter Type	**Name**	**Nominal Value**	**Range**	**Units**
Uncertain Parameters	I_{yy}	1	± 0.2	kgm^2
	I_{zz}	1	± 0.2	kgm^2
	byz_{coup}	0	± 0.1	Nm
	bzy_{coup}	0	± 0.1	Nm
	ω_s	$5\frac{\pi}{180}$	$\pm 0.5\frac{\pi}{180}$	$rad.$
	F_{max}	400	± 40	N
	α	π	$\pm$pi	$rad.$
Configuration Parameters	I_{xx0}	0.5	± 0.3	kgm^2
	x_{com0}	6.25	± 5.75	m
	I_{mib}	1000	± 500	kgm^2
Dependent Parameters	I_{xx}	I_{xx0}	± 0.1	kgm^2
	x_{com}	x_{com0}	± 0.1	m

The next section briefly formalises the performance function employed in this study.

6.1.3 Performance function

As previously stated, the performance function considered in this study is the total number of thruster actuations, which may be written as follows:

$$J(\delta) = size(N_2 \neq 0) + size(N_3 \neq 0) \tag{6.6}$$

Since N_2 and N_3 are control vectors consisting of impulses corresponding to each thruster actuation, computing the size of their individual non-zero elements and adding them together gives the total number of thruster actuations corresponding to parameter vector δ.

The performance function value for the nominal plant is 130 actuations. A previous study [121] used a combination of Monte Carlo and optimisation to find the performance for a previous version of the controller. The worst case performance found was 202 thruster actuations, and is considered rather high. The performance criterion is set as $J(\delta) \leq 200$ actuations in this analysis. The probabilistic criterion is stated as in Equation (2.23), i.e.,

$$P[J(\delta) \geq 200] < p_{critical} \tag{6.7}$$

where $p_{critical}$ is 10^{-6}, decided in consultation with the control design team. What follows in the next section is an account of the performance analysis carried out on the barbecue mode attitude controller, considering the parameters specified in Section 6.1.2.

6.1.4 Results

Before carrying out the analysis, the CE code was integrated into WCAT-II. This provides a single framework for industry partners to carry out optimisation-based and probabilistic-based analysis. Turning now to the barbecue mode performance analysis, it is noted that an evaluation of the performance function involves setting the uncertain parameter values as dictated by the WCAT-II algorithm (DE or CE), simulating the closed loop model in Simulink, and computing the total number of actuations. A single evaluation takes about 12 seconds. A DE run is set up with 100 iterations, and 25 population members (all of the settings are identical to those in Table 5.2). The resulting worst case performance is 406 actuations, and is found in the 68^{th} iteration of DE. The evolution of the maximum and mean costs of the DE population in every iteration is shown in Figure 6.4.

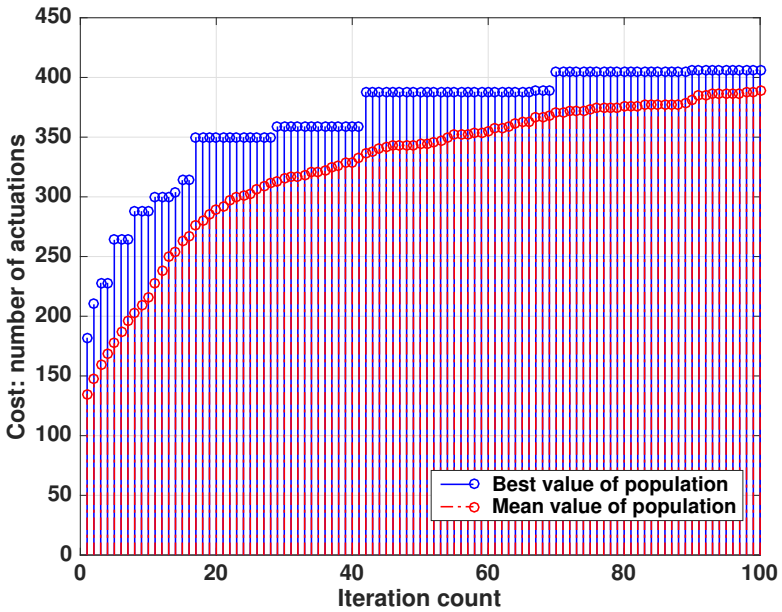


Figure 6.4: DE performance value evolution as iterations progress.

Clearly, 406 thruster actuations is a very serious violation of the performance criterion that places a safety bound of just 200 actuations. In order to understand how severe the threat of such a violation is, it is vital to determine the probability associated with this worst case value. It is also desired to find the probability associated with violation[1] of bound, i.e. $P[J \geq 200]$. A CE run is set up according to the parameters in Table 5.3. In this particular problem, a fairly empirical approach is needed to arrive at the performance levels γ at which probability is measured. Since in general, it is interesting to study the curve starting from a probability of 1 going down to a probability of 0, the performance levels are to be chosen appropriately. Finding the upper limit of γ is fairly straightforward — the figure closest to the DE worst case is selected. The lower limit of γ is typically chosen close to the nominal performance, however, that provided probability values lower than 1. It is found that starting at $\gamma = 60$ yields a visually complete probability profile of performance, which is shown in Figure 6.5.

The first observation is that the minimum number of actuations is always above 80. Secondly, there is a drastic fall in the probability while going from 180 actuations to 190 actuations. While the probability of achieving 180 actuations is 0.122, the probability of achieving 190 actuations is of the order 10^{-8}, which is a rare event. Thirdly, the probability decreases gradually from there until $\gamma = 410$, when the probability is of the order 10^{-25}. This curve, therefore, constitutes a long tail distribution. Because of such behaviour, the curve fitting does not create an elegant curve like those in Chapter 4. The probability values associated with the performance levels γ are provided in Table 6.2.

The most important observation is that $P[J \geq 400]$ is small, of the order 10^{-19}, which constitutes a rare event. Therefore, the worst case performance found by DE, although

[1] Such a measure is called a failure probability in some applications[99]

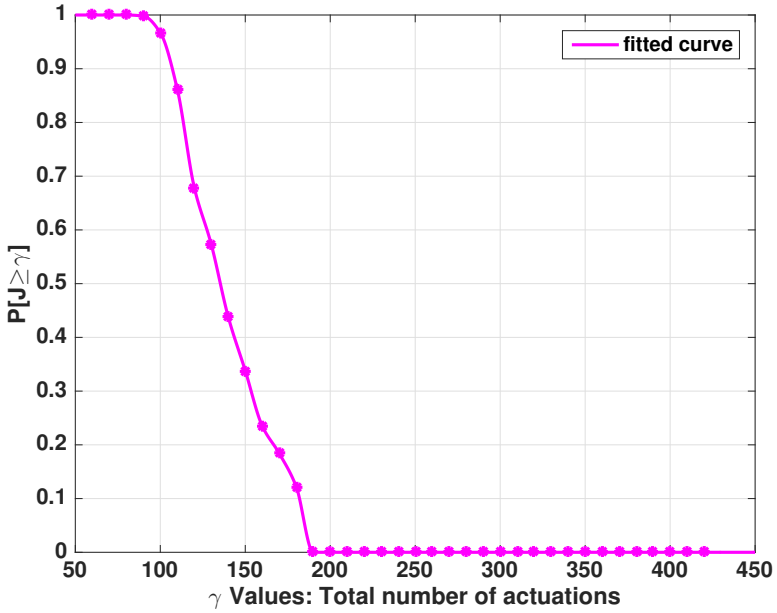


Figure 6.5: Probability profile of performance for attitude controller of long coasting upper stage

alarming, has a very low likelihood of occurrence. Interestingly, CE computes a higher worst case performance, at 418 actuations. CE $P[J \geq 410]$ is of the order $\times 10^{-25}$[1]. These small numbers reinforce that CE is effective in estimating probabilities of extremely rare events.

In addition, Table 6.2 reveals that the probability of violating the performance bound is $P[J \geq 200] = 3.12 \times 10^{-8}$, which is a rare event. This is a significant result, and proves that the barbecue mode controller is probabilistically robust, as per the criterion in Equation (6.7). In contrast, robust performance in the classical deterministic sense is not satisfied. This forms the crux of the conservatism issue as pertains to worst case analysis techniques. Practitioners are normally only concerned with events that occur with a probability of 10^{-6} or greater, and so decision makers may accept such a risk

[1] One interpretation of such a probability is that a classical Monte Carlo run would need at least 10^{25} samples to arrive at a performance greater than 410 actuations.

Table 6.2: Probability values associated with various performance levels

γ	60	70	80	90	100
$\mathbf{P}[\mathbf{J} \geq \gamma]$	1	1	1	0.99	0.97

γ	110	120	130	140	150
$\mathbf{P}[\mathbf{J} \geq \gamma]$	0.86	0.68	0.57	0.44	0.34

γ	160	170	180	190	200
$\mathbf{P}[\mathbf{J} \geq \gamma]$	0.24	0.18	0.12	3.12×10^{-8}	2.18×10^{-8}

γ	210	220	230	240	250
$\mathbf{P}[\mathbf{J} \geq \gamma]$	1.67e-08	1.23×10^{-8}	1.06×10^{-8}	2.78×10^{-9}	4.15×10^{-9}

γ	260	270	280	290	300
$\mathbf{P}[\mathbf{J} \geq \gamma]$	6.97×10^{-9}	2.1×10^{-9}	2.34×10^{-9}	1.56×10^{-9}	7.95×10^{-10}

γ	310	320	330	340	350
$\mathbf{P}[\mathbf{J} \geq \gamma]$	2.8×10^{-10}	1.11×10^{-10}	$.50\times10^{-11}$	2.48×10^{-11}	1.08×10^{-11}
γ	360	370	380	390	400
$\mathbf{P}[\mathbf{J} \geq \gamma]$	7.49×10^{-12}	2.87×10^{-12}	1.01×10^{-13}	2.24×10^{-14}	3.43×10^{-19}

γ	410	420
$\mathbf{P}[\mathbf{J} \geq \gamma]$	3.90×10^{-25}	0

of violation. Interestingly, the CE analysis both confirms and rejects the worst case performance found by DE.

The uncertain parameters associated with poor performance are now investigated. The final mean and standard deviation values μ_T and σ_T at each γ are shown separately for uncertain parameters in Figure 6.6, for the configuration parameters in Figure 6.7 and for the dependent parameters in Figure 6.8. One striking observation in all the three cases is that the standard deviation values oscillate for the initial few γ values. Interestingly, the μ_T values of all parameters are close to their initial values of 0 when this occurs. Such an effect has not been observed in other studies, and may be attributable to the fact that the performance value takes only integer values.

Since at the final values of γ, σ_T is very low (often smaller than $\sigma^* = 10^{-6}$), the dis-

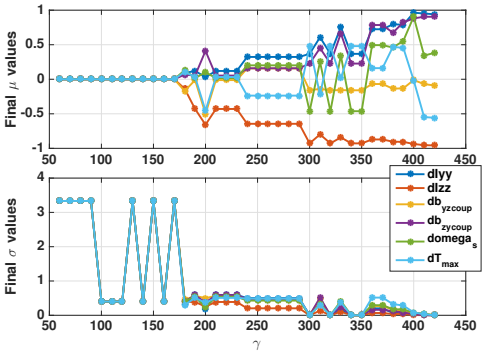


Figure 6.6: Final CE mean and standard deviations of uncertain parameters for upper stage barbecue mode

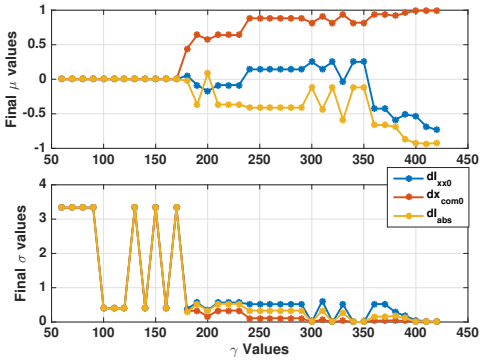


Figure 6.7: Final CE mean and standard deviations of configuration parameters for upper stage barbecue mode

cussion in the rest of this section treats the μ_T values at $\gamma = 420$ as the worst case parameter values. With regards to the worst case parameter values, some clear trends are observed in Figure 6.7. x_{com0} settles at its extreme value. This may be readily explained, because when the lever arm is high, the value of torque is high, and so the number of actuations needed will be high. In addition, I_{abs} is at its minimum, and since it is inversely proportional to Mib, Mib is high. Then the torque threshold increases. This is counter-intuitive, but could trigger a compound effect and increase the actuations. The nominal longitudinal inertia I_{xx0} is also close to its minimum. These parameter regions are clearly problematic, and this behaviour may also be observed in the heat plots 6.9 and 6.10.

The uncertain parameter final mean values in Figure 6.6 reveal that the two lateral inertias I_{yy} and I_{zz} tend to the extremes in opposite directions, with I_{yy} being close to its maximum and I_{zz} close to its minimum. In addition, the spin rate frequency tends towards the maximum (although the μ_T evolution appears to oscillate as γ increases). This is expected, because the nutation frequency is proportional to the spin frequency, and higher torques need to be generated when nutation is higher.

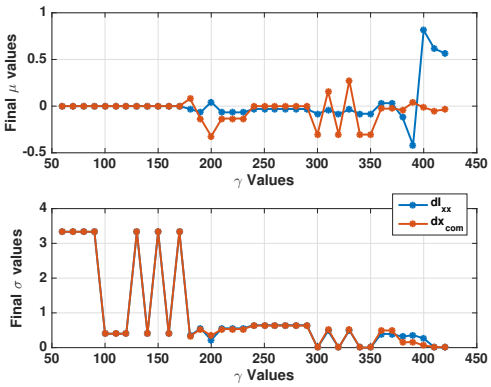


Figure 6.8: Final CE mean and standard deviations of the configuration-dependant parameters for upper stage barbecue mode

With regards to the dependant parameters, the longitudinal inertia component remains close to its nominal value for most performances, and increases towards its maximum close to the worst case performance. The lever arm remains close to its nominal. Recall that the nominal values of these two parameters are themselves configuration parameters, and varied by CE. The lever arm uncertainty x_{com} remaining close to its nominal value implies that the configuration value x_{com0} is already close to its worst case.

A deeper insight into how the parameter regions impact the performance values may be achieved from the heat plots in Figure 6.9. These heat plots are created by scattering each parameter's IS run samples, and plotting against the performance values obtained. Since there are 11 parameters, the data is plotted taking 2 parameters at a time, with the z-axis showing the performance obtained. In Figure 6.10, the plot is collapsed to two dimensions. Because the colours of each sample are mapped to the performance value, this 2 dimensional plot retains all the information from Figure 6.9, while revealing a clearly how each parameter varies.

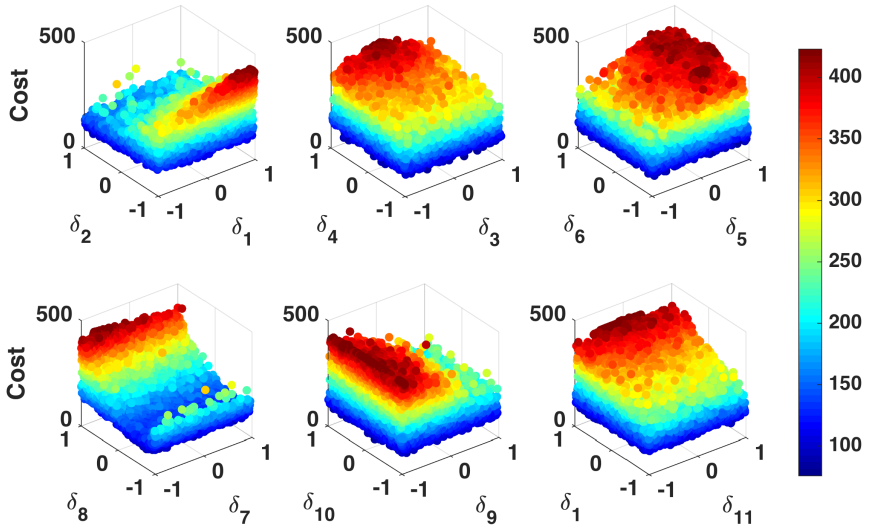


Figure 6.9: Heat plot of the number of actuations, taking into account two uncertain parameters at a time.

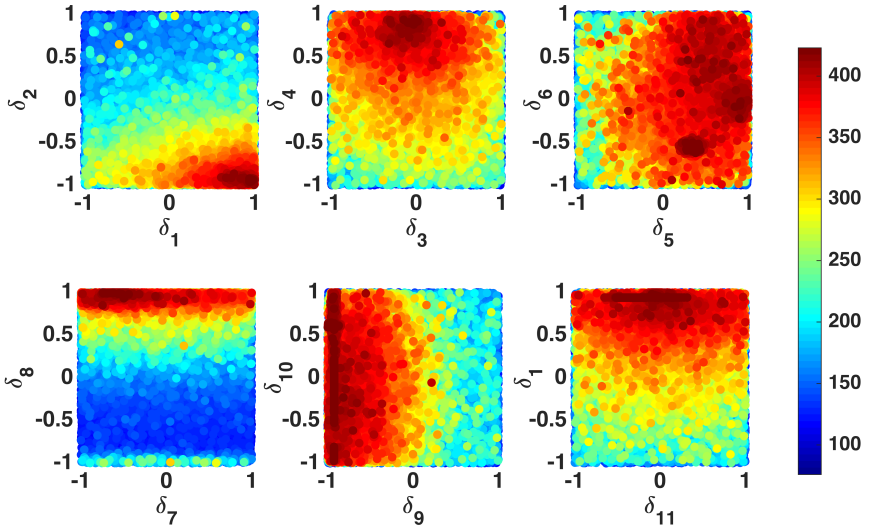


Figure 6.10: Heat plot showing XY views

The first plot in Figure 6.9 shows the two lateral inertias I_{yy} and I_{zz} plotted against the performance. As previously observed, the performance gets degraded as I_{yy} approaches 1 and I_{zz} approaches -1. An observation from the corresponding plot in Figure 6.10 shows that the performance is safe (less than 200 actuations) in a large portion of the parameter space. However, the other five plots show a larger proportion of the parameter space being covered by larger performances. These plots appear to have a large probability associated with bounds violation. Recall, however, that the probabilities associated with achieving such performances are very small (smaller than 10^{-7}), and these high performances are attributable to CE sampling in a biased way in problematic regions in order to find the worst possible performance values. The probabilities finally computed by CE are scaled back via the likelihood ratios, and these probabilities reflect accurately the true probability of violation. As mentioned in Chapter 4, such plots may be plotted in a lucid and straightforward way using data from CE's final runs, because CE samples from the poor performance regions of the parameter space rather densely.

This concludes the section on the attitude control system of a long-coasting launcher upper stage in the barbecue mode. This section described the dynamics of the upper stage in the barbecue mode, and briefly presented the designed controller, the parameters considered, and the results of its performance analysis using CE. This study was significant in that the conservatism of worst case performance was quantified. Although the worst case was a serious violation of the performance criterion, the probability associated with such a performance is very low.

In the following section, CE is implemented on a far more complicated launcher model, i.e. the Vega launcher model[124]. This analysis is carried out in the post-design stage on a high-fidelity simulator[1].

[1] As a result, the treatment is not as transparent as in the barbecue mode

6.2 VEGA Launcher Model

Vega is a four stages, multi-payload, expendable launch vehicle developed by European Launch Vehicle, Italy under the contract of ESA. The commercial debut of the launch vehicle, shown in Figure 6.11, occurred in February 2012. The launch vehicle is 30m in length and 3m in diameter, and is meant to cater to the small launcher requirement in industry. In this study, carried out in the frame of the research project "Robust Flight Control System Design Verification and Validation Framework" (ESA AO/1-6322/09/NL/JK), the launch vehicle controller performance is analysed. Previously, deterministic worst case analysis techniques were assessed by measuring the validation gap between traditional Monte Carlo outcomes and worst case analysis results. Subsequently this research activity aims at narrowing down the previously determined gap by using probabilistic worst case analysis techniques while associating tight probabilistic risk levels to the requirements under investigation. The project seeks to industrialise promising approaches to $V\&V$, emphasising that there is no room for errors due to reduced testing. The project is contracted by ESA, and is carried out in collaboration with ELV, University of Bristol and University of Exeter.

The primary objective of the project is systematic uncertainty quantification. The analysis aims to occur at the intersection of TVC performance assessment, and loads analysis [1]. The aerodynamic load must be analysed since the structure of the launch vehicle, and therefore the mission success, depend on it. In this study, Vega's 1st stage, known as the P80[2], is analysed. The main functions of P80 are the lift off and pitch over, the gravity turn (where the maximum dynamic pressure is experienced) and the first separation event. The first flight phase, i.e. the atmospheric flight phase from 35km and 60km is considered.

[1] which typically has its own rigorous analysis process with structural and fluid mechanics engineers
[2] so called because the first stage carried 80 tonnes of propellant in the early versions

Figure 6.11: Vega launcher. Source: ESA

The controller design is challenging because Vega is inherently aerodynamically unstable in the atmospheric phase [125]. The centre of pressure is above the centre of gravity. The system parameters such as the aerodynamic coefficients, bending mode effects and the crucial Mass, Centre of Gravity and Inertia (MCI) parameters vary significantly throughout the launch timeline. This is because flight and experimental data is lacking. The TVC must therefore ensure stable response while not deviating from the reference trajectory.

The major challenges of P80 control in atmospheric flight are:

- Significant wind gust disturbances
- The mechanical loads are to be limited due to structural issues
- Roll coupling [125]
- Control minimisation
- High level of scattering and variation of parameters
- Coupling control with the elastic modes

- Following guidance command under the above challenging conditions.

The control architecture is a Thrust Vector Control (TVC) architecture, as is typical for launch vehicles. The control law is a PID controller with gains scheduled over all four stages to accommodate changing flight conditions [126]. The PID gains on the attitude are designed for rigid body performance and stabilization, but also to provide large frequency separation between the rigid body and the first bending mode frequencies. For the P80 phase the bending filters are designed to phase stabilize the first bending mode and gain stabilize the upper modes. Due to axial symmetry of launchers, the control is assumed to be the same in the yaw and pitch channels. A coupling term between pitch and yaw is introduced to account for roll coupling,and this is done via gyroscopic compensation. As already mentioned, the controller gains are scheduled over the flight phases to accommodate variations of flight parameters, especially in P80 phase where dynamic pressure, MCI, thrust and structural properties vary greatly.

The actuators are 6 thrusters with on/off responses. The tuning parameters are the PID and PD controller gains, and the actuator MiBs [126]. Basic controller $V\&V$ tests are carried out in the frequency domain during the design. This consists of checks on low and high frequency gain and phase margins, as well as Nichols plots, to ensure that nominal stability and performance is satisfied. The performance must then be analysed in the time domain. The current study aims to assess the robust performance of the TVC controller. It is a post-design $V\&V$ and is carried out on a high-fidelity nonlinear simulator in the time domain. The study seeks to answer the following questions, many of which overlap the overall study questions:

- What performance values occur for "most" of the cases?
- What is the probability of violation of the performance bound?
- How abrupt is the degradation towards poor performance?

- If violations exist with a non-zero probability, how much of the performance requirement may be relaxed?
- What region of the parameter space is problematic?

Since the Vega is a commercial vehicle, it has already undergone significant pre-flight and post-flight analysis [127],[128], [129]. In the study [129], an MC analysis is carried out on a simplified LTI system, and the analysis points to the performance bound being fulfilled. However, a $\boldsymbol{\mu}$-analysis on the same system shows violations [125]. All of these studies repeatedly highlight the need for more reliable techniques. Studies also emphasise that the worst cases computed must not be conservative, and must not direct onerous redesigns when not necessary. Even controller gain retuning is justified only if the probability of occurrence is not too low. This is an important practical consideration because resources and launch constraints often are unavailable, and delaying a launch for retuning purposes is not usually feasible. This knowledge will aid ESA decision makers by giving them quantitative means for relaxation of performance bounds when justified. This essentially shifts the role of decision makers from before the analysis (expert knowledge elicitation etc) to after the analysis, where the physical interpretations of worst case combinations can be analysed with their system expertise. The aim is also to improve and shorten the current $V\&V$ process for GNC. The focus is on reconciling the probabilistic and deterministic nature of the parameters.

An initial V&V campaign [126] uses Monte Carlo simulation to identify worst cases. A large number of samples, 2×10^5, are evaluated. It identified a performance criterion (Q*alpha) that is particularly critical. An inference drawn in the previous study is that MC is good for finding empirical mean and variances, but potentially unreliable for computing extreme values. The study also identified a small parameter set consisting of 4 uncertainties that is particularly important. Additionally, a separate emphasis is

on the impact of winds on performance. Particularly for $Q\alpha$, wind is found to impact the cost significantly. Therefore, the analysis is divided into two parts: the first part assesses the four uncertain parameters and their effect on performance, while the second part assesses the impact of wind on performance. Before proceeding to the performance analysis, details on the high-fidelty simulation model of Vega, i.e., VEGACONTROL are presented.

6.2.1 Description of the VEGACONTROL3 model

The ELV VEGA CONTROL simulator is a Matlab C-coded simulator used in industry to prepare and validate the VEGA fight management and flight control system. The model includes the full set of non-linear equations of motion. The simulator includes a non-linear model of the eletro-mechanical actuator dynamics with associated backlash and delays. The simulator also implements a QUASAR Inertial Sensor Unit with its noise and bias characteristics. The propulsion model reflects the dynamics of the P80 solid propulsion system with validated thrust curves that include thrust oscillation effects to assess proper execution of the separation dynamics. The flight mechanic model assembles a fully validated non-linear aerodynamic model for normal force, drag and tree axis moments curves that depends on Mach and angle of attack.

Kinematic coupling in all axes are implemented to reflect pitch yaw roll couplings. A high fidelity structural flexible mode model describing the launcher deformation is included to assess proper filtering and stability properties. The atmosphere model includes also a set of measured wind-gust input models representative for the launch site Kourou. The launcher dynamics are driven by the FPSA ADA/C-flight code reflecting the flight management system for the time line sequence command and execution of associated guidance navigation and control system for Thrust Vector Control (TVC) and roll and attitude control and other support functions such as acceleration threshold

detection and pyro-valve command for stage separation.

Note that the core controller and simulators are p-coded, and the verification team has no access to any internal workings. Only the uncertain parameters may be manipulated, and the Q-alpha is plotted. This is a black-box philosophy which creates a separation between design and analysis so that no inadvertent assumptions are made. The Simulink model of Vega launcher with flexible modes is shown in Appendix B.

6.2.2 Performance Criteria and Uncertain Parameters

The main requirement is the load Q-alpha, which is particularly important for the first stage as it performs a gravity turn[1]. The criterion for Q-alpha is dynamic, and varies with the Mach number. In the atmospheric flight phase the $P80$ accelerates upto 5.6 Mach. The maximum structural loads that the launch vehicle can withstand as the Mach number increases are determined experimentally [126] and shown in the form of a safety envelope in Figure 6.12. The critical phase, i.e. the gravity turn, occurs with the first 20 seconds after lift-off. The gravity turn manoeuvre is newly designed for every mission, so the worst case is mission dependent.
The criterion is also defined probabilistically by the consortium. It is stated that $Q\alpha$ must be within the safety envelope for 99% of the cases, i.e., $P[J > \gamma_{des}] < 0.01$, where γ_{des} is dynamically defined by the safety envelope. This criterion is verified in [126] with 3700 MC runs, with a synthetic wind included. With these conditions, the performance is satisfied for 100% of the 3700 cases.

Two uncertainty sets are considered. Table 6.3 shows a larger uncertainty set including some critical uncertainties (identified through system expertise). The relative impor-

[1] the launch vehicle manoeuvre after the initial vertical ascent, in order to reach a horizontal position at burnout, the vehicle has to turn and this is automatically done by dynamics, in what is also sometimes called the pitch over manoeuvre

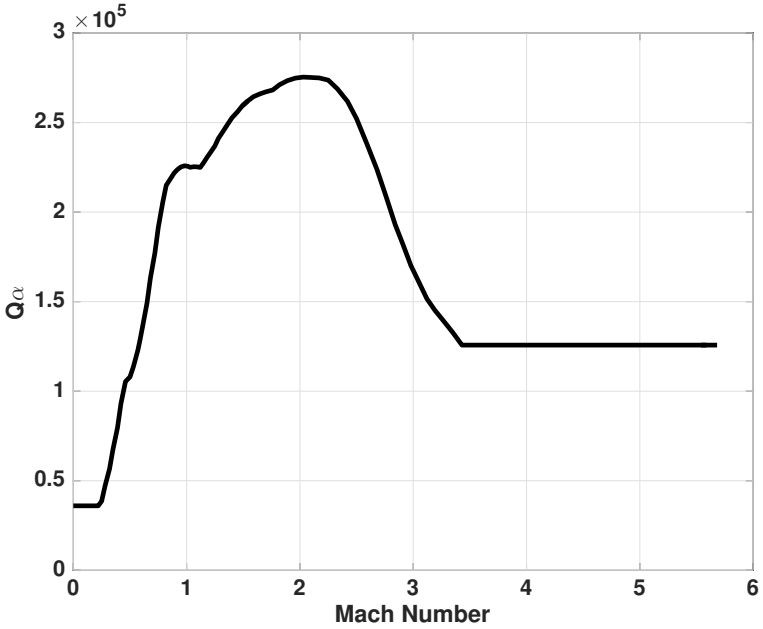


Figure 6.12: Safety envelope for $Q\alpha$ w.r.t Mach number

tance of these uncertainties is found to be high using methods. In the second analysis where wind effects are to be considered, a larger uncertainty set is considered, with aero parameters etc.

Note that the exact parameter nominal values and ranges are in `p-code` files and not accessible. All of the parameters are normalised in [-1,1], and CE is interfaced with the algorithm through a set of FLAGS. Each uncertain parameter has an associated flag, which CE can assign a value to. The first set includes only four of the parameters, two of which are listed in the table, viz., dTc, $air\_density\_scat$ and two that are not - they are the roll torque contribution by the solid rocket motor SRM_roll and the lateral mounting misalignment of the inertial reference system $IRSMounting\_Y$[1].
The analysis accounts for real wind models. A brief description of these models is presented in the following section.

[1]The variable names of the VEGACONTROL model are preserved.

Table 6.3: Uncertain parameters in Vega P80 model considered in the CE analysis.

Parameter Type	**Name**	**Physical Meaning**
Aero and Atmospheric	unc_CN, $disp\_CN$	Normal lift force coefficient, and associated dispersion
	unc_CA, $disp\_CA$	Axial coefficient, and associated dispersion
	unc_Xcp, $disp\_Xcp$	Longitudinal centre of pressure coordinate, and associated dispersion
	$air\_density\_scat$	Air density
Propulsion	$dIsp$	Specific impulse
	dTc	Burn time
TVC	TVC_SF_A, TVC_SF_B	Thruster gains
	TVC_bias_A, TVC_bias_B	Thruster biases
	TVC_mis_A, TVC_mis_B	Thruster misalignments
MCI	$P80dM$	P80 Total mass
	$P80dM_prop$	P80 Propellant mass
	$P80dJx$, $P80dJy$, $P80dJz$	P80 Moments of inertia
	$P80dJx\_S$, $P80dJy\_S$, $P80dJz\_S$	P80 Structural (non-rigid) moments of inertia
	$P80dxCOG$, $P80dzCOG$	Centre of gravity
	$P80dxCOG\_S$, $P80dyCOG\_S$, $P80dJzCOG\_S$	P80 Structural (non-rigid) Centre of gravity
	$PVP\_offsetZ$	TVC pivot point lateral offset

Table 6.4: Wind models included in analysis of Vega launcher P80

	Wind file name
Wind model 1	01_1993_09_12_005.wind
Wind model 2	02_1991_26_12_005.wind
Wind model 3	07_2005_03_12_004.wind

6.2.3 Wind descriptions

In order to account for real winds (rather than synthetic ones), a set of wind models is available in VEGACONTROL. The names of the wind models included in the analysis are listed in Figure 6.4. The wind files are recorded on particular days, and at every attitude the wind magnitudes, gradients and directions are saved as a .wind file which is accessible. The wind-gust input models are representative of the launch site Kourou. The development of these wind models is detailed in [130].

This section has so far presented details on the simulation model, the controller, the uncertain parameters and performance criteria. The next section now details the two part performance analysis for the Vega P80 stage in its atmospheric flight phase.

6.2.4 Results

The implementation of CE on VEGACONTROL is now detailed. The first part of the analysis considers the reduced uncertainty set consisting of four parameters, i.e., $IRSMounting\_Y$, SRM_roll , dTc, and $air\_density\_scat$. The probability profile of performance is generated, with the CE parameters as shown in Table 6.5. Observe that at each iteration of CE, $N = 50$ performance function evaluations are allowed, and $N_1 = 100$ performance evaluations are allowed in the final IS run. In order to find the probability in such a way that high computation costs are not incurred, the computational budget is reduced. Contrast this with Chapter 5, where 500 evaluations are allowed each time for the linearised iGNC system. Such a reduction in computational resources is justified because each performance function evaluation takes about 7.8 sec-

Table 6.5: CE Algorithmic parameter values for Vega launcher analysis

CE Parameter	Parameter Description	Parameter Value
N	Evaluations at each CE cycle	50
N_1	Final IS run evaluations	100
ρ	Rarity factor	0.1
t_{max}	Maximum number of CE iterations allowed	30
σ^*	Threshold standard deviation	10^{-6}

onds. The industry partners desire an analysis that can be completed in the matter of a few hours ,rather than a few days (assuming $N = 500$, $N_1 = 500$ evaluations for each γ and a modest 10 values of γ, the time taken for computation would be about 1 day and 5 hours, which is an optimistic estimate since it does not account for additional CE iterations to update the IS distributions).

Each evaluation of the Vega simulator takes about 7.8 seconds on a Windows computer. The CE initial distributions of these 4 uncertainties are Gaussian distributions with mean 0 and standard deviation 0.4, truncated at [-1,1]. The standard deviation chosen is such that the selected region corresponds to approximately 98.5% of the entire Gaussian (2.5σ). The rationale for such a choice of μ_0 and σ_0 is to obtain a distribution with nominal value at 0 and such that a large number of values lie in the interval [-1,1]. The probability estimation algorithm is applied to find $P[J(\delta) \geq \gamma]$ for every 10,000 $Padeg$ of $max(Q\alpha)$ from $1.8{*}10^5$ $Padeg$ onwards, until a probability of 0 is encountered.

The probability profile of performance generated by CE is as shown in Figure 6.13, along with the final mean and standard deviations in the importance sampling distributions at each value of γ. At the start of the curve, i.e., at performances of 1.8, 1.9 and 2×10^5 Padeg, the associated probability is 1. This indicates that the maximum of $Q\alpha$ will always be at least $2 \times 10^5 Padeg$. From here, the probability gradually decreases until a probability of zero occurs at $3.1 * 10^5 Padeg$. The tail end of this probability profile is especially interesting, as it provides information about the worst case and

performances close to the worst case. A small probability of $1.13 * 10^{-6}$ is observed to occur at 3×10^5 $Pa.deg$. We can infer, therefore, that the worst case value of $max(Q\alpha)$ lies between $3 * 10^5$ and $3.1 * 10^5 Padeg$, which is clearly a violation of the maximum allowed value of $Q\alpha$, i.e., 2.75×10^5 $Padeg$. In order to further narrow this region, the algorithm was applied to the performance levels 3.06×10^5 and $3.07 \times 10^5 Padeg$. These cases gave probabilities of 1.044×10^{-7} and 0 respectively. The true worst case may therefore can be said to lie in $[3.06 \times 10^5, 3.07 \times 10^5] Padeg$. Further, the probability of its occurrence is a rare event, i.e., one with a probability lower than 10^{-6}. This agrees with the expectation that the worst case be a rare event.

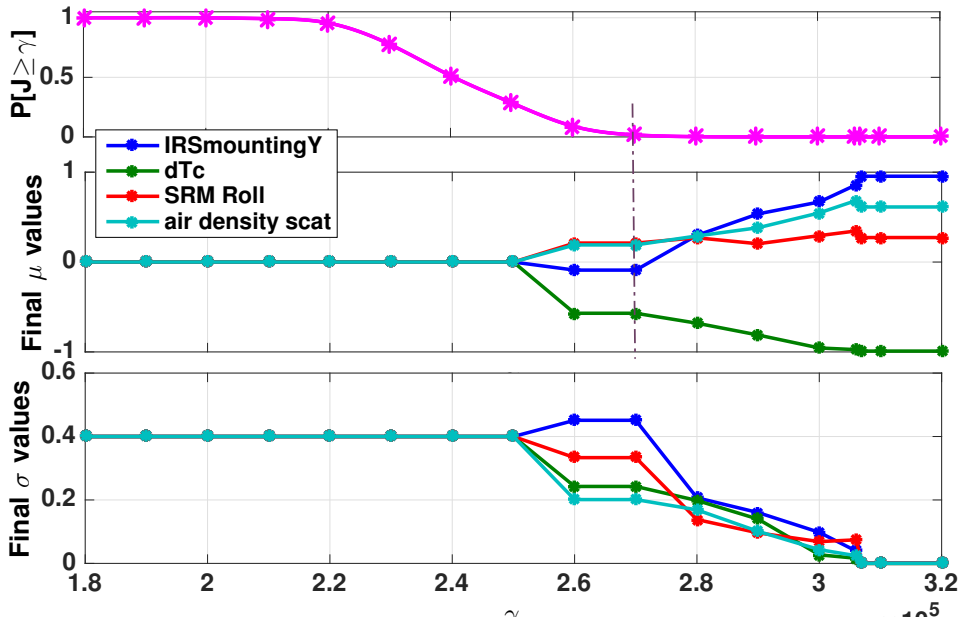


Figure 6.13: Probability profile of performance shown with the final mean and standard deviations as γ increases.

Apart from the worst case probability, the probability profile offers information on the intermediate performance levels between 2 and $3*10^5 Padeg$, corresponding to the decay region in Figure 3.3. As the performance degrades, the probability gradually reduces.

The spline fitted curve in Figure 6.13 is useful in determining probability values at any performance level of interest. With regards to the performance bound $\gamma_{des} = 2.75 \times 10^5$ $Pa.deg$, the PPoP curve reveals that the associated probability is 0.006, which is lower than $p_{critical}$. Hence, although deterministic robustness is not satisfied due to bound violation, probabilistic robust performance is satisfied. This also serves to identify how realistic a performance requirement is. In the current scenario, for example, imposing that the maximum dynamic pressure must be lower than $2.3 * 10^{-5} Padeg$ is not a good choice, since there is a 78% probability that the requirement will be violated. A good requirement is one that has a small probability of violation.

The following plots show some results for $\gamma = 3.06 * 10^5$ $Pa.deg$, in order to gain insight into the parameters that contribute to such a large performance. Figure 6.14 shows the final pdfs of the four parameters, optimised by the cross entropy algorithm. Observe that the resulting variance has been reduced significantly as compared to the initial value.

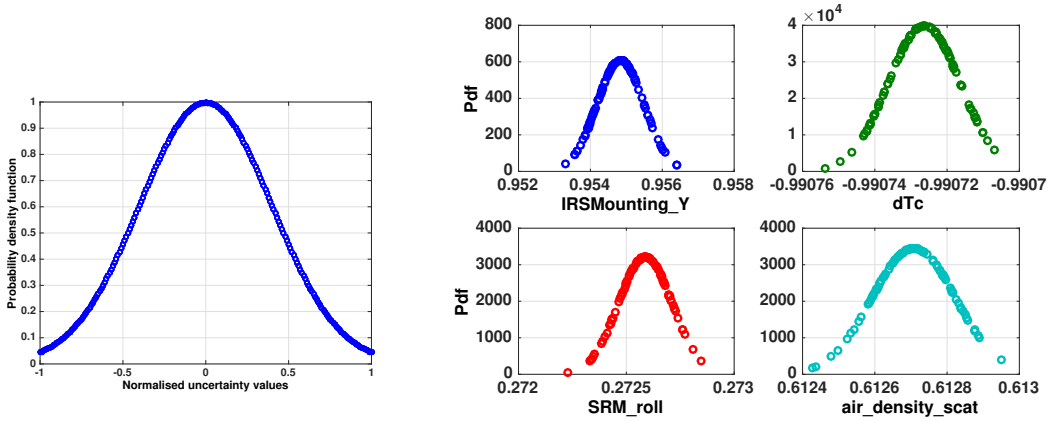


Figure 6.14: Initial and final pdfs of the four uncertainties

Figure 6.15 shows the tail of the probability profile of performance, corresponding to

the PPoP critical region in Figure 3.3, along with the means μ_T of the four uncertain parameters shown as bar plots. The bar plots provide a clear sense of the directionality of the parameters that lead to poor performance. It is also interesting to study how $Q\alpha$ varies with the Mach number, and also with γ. Figure 6.16 shows how $Q\alpha$ varies with the Mach number over the 17 γ levels. The scatter plots in Figure 6.17 show the performance distribution as well as the distribution of each of the uncertain parameters.

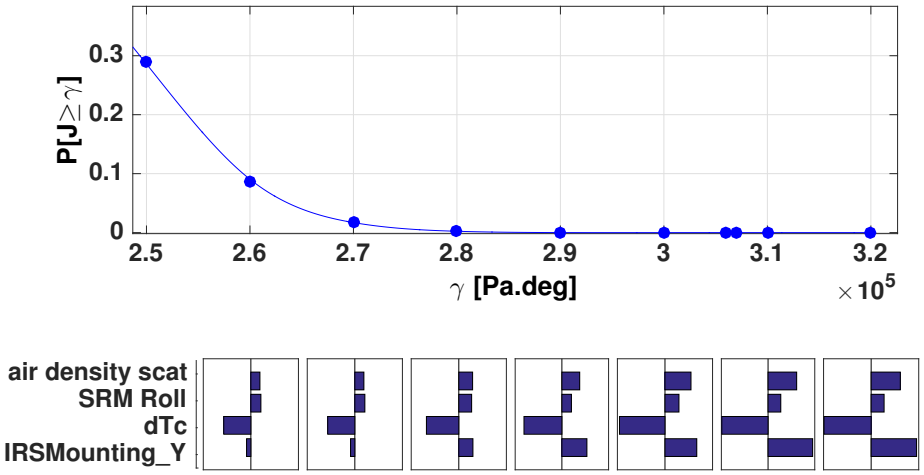


Figure 6.15: Probability profile critical region, shown with final means of the uncertain parameters

The second part of the analysis focuses on the effect of winds on the probability profile. It is expected that there is no violation of bounds in the absence of wind, and that some wind profiles affect the performance more adversely than others. The winds considered are representative winds measured at various altitudes on different dates. An in depth explanation is in [130].

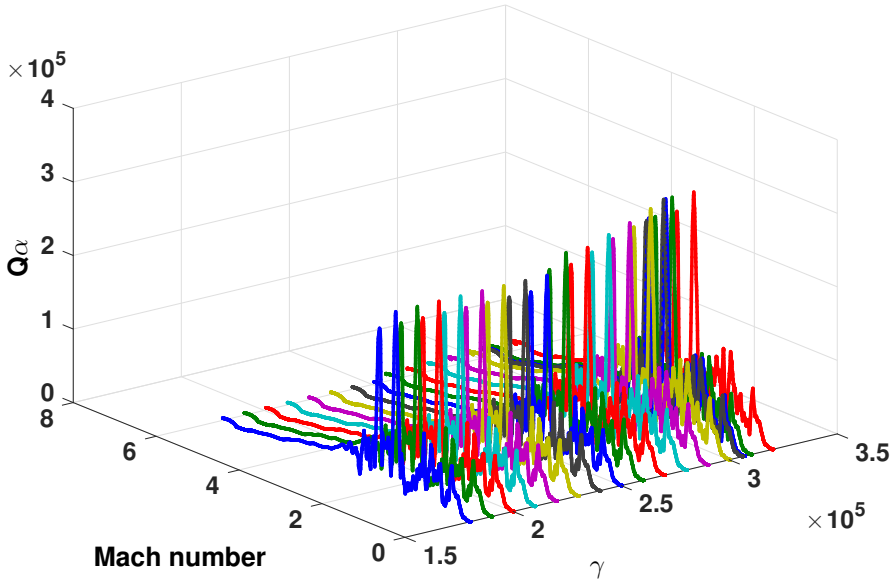


Figure 6.16: Qalpha vs Mach as gamma varies

Before proceeding, the cost function is modified to the ratio $\frac{Q\alpha}{Q\alpha_{max}}$. This makes it more convenient to compare the probability profiles of performance for all wind cases. It also makes the cost function static, and more convenient to detect violations of the bound.

The probability profile of performance is generated for each wind model, one at a time, using the CE parameters in Table 6.5. The four curves representing the probability profiles of performance are shown in Figure 6.18.

Table 6.6 shows some salient data regarding the tail performances for all four wind cases. The no wind case shows there are no instances of performance exceeding 55% of the safety envelope, and so the system satisfies robust performance. This is also clear from the $Q\alpha$ versus Mach number plot shown in Figure 6.20. However, it is practically impossible to have the launch vehicle subject to no winds in the entire atmospheric

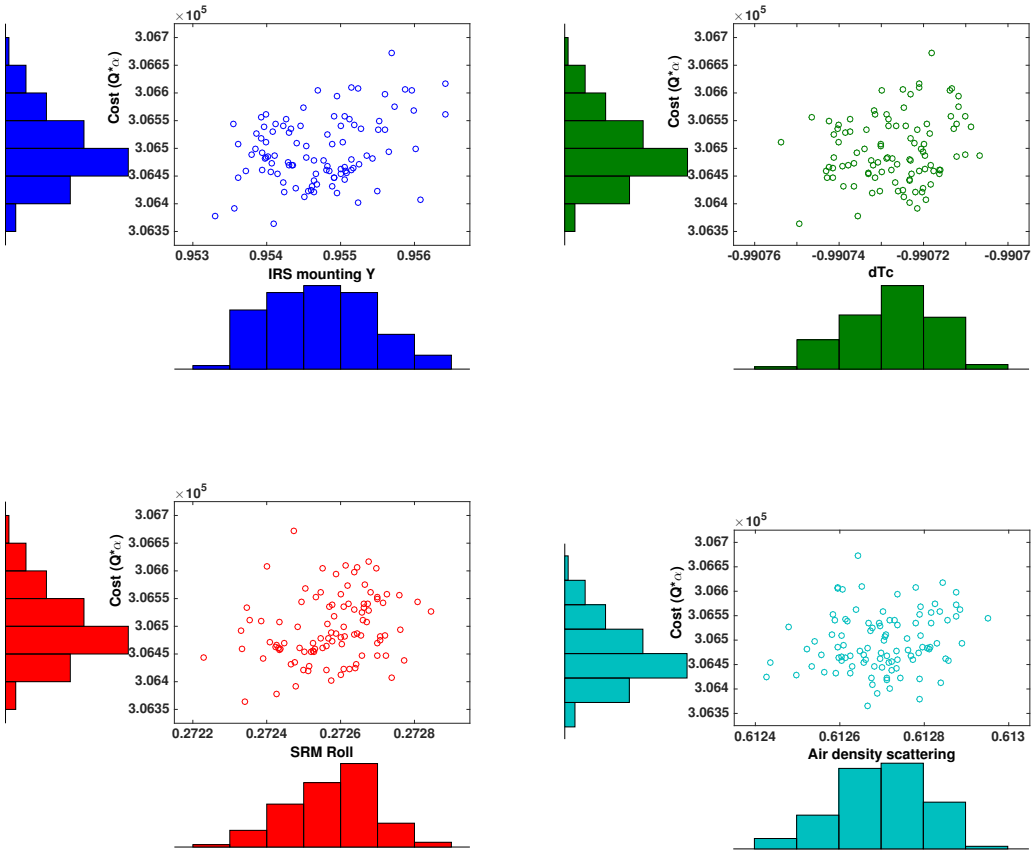


Figure 6.17: Final run with the importance sampling parameter distributions. The costs obtained at different values of the uncertain parameters, along with the parameter and performance distributions are shown here.

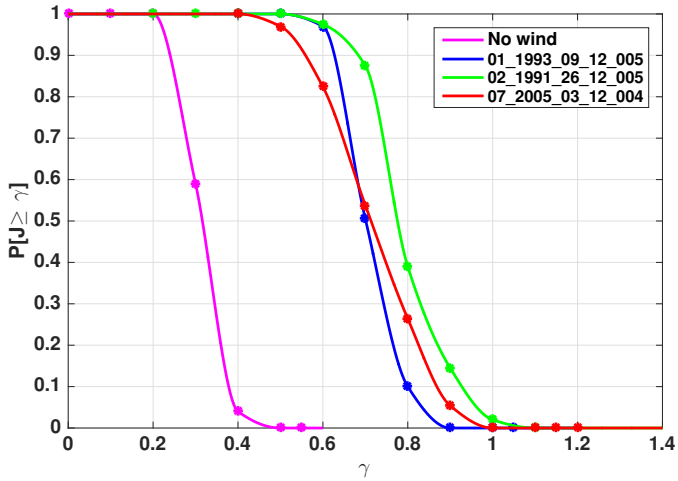


Figure 6.18: Probability profiles of performance for all wind cases and no wind case

flight time line, so the other three wind model analyses are crucial. Further, the worst case uncertain parameters associated with each wind case are shown in the bar plots in Figure 6.19.

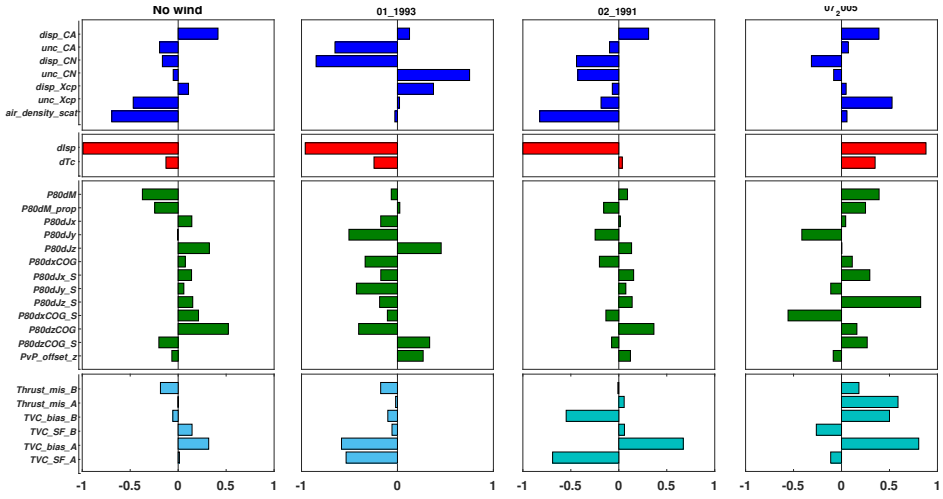


Figure 6.19: CE worst case uncertain parameter combinations for all wind cases

Considering wind model 1 (i.e., 01_1993_09_12_005), the safety bound is violated with a probability in the order of 10^{-24}. The CE worst case (i.e., the maximum performance

found in the final importance sampling run) shows about a 2% violation of the safety bound, and this violation occurs at around 2.5 Mach, as seen in Figure 6.21. Although the probability associated with this violation is small, the magnitude of the structural load is high, at 2.55×10^5 $Pa.deg$. The second bar plot in Figure 6.19 show the uncertain parameter combinations that lead to this high performance. It is concerning to note that several of the MCI parameter values lie in the middle of the parameter region, close to their nominal values. This implies that even small deviations in this crucial class of parameters, when combined with other effects, could lead to poor performance.

With regard to wind model 2 (i.e., 02_1991_26_12_005), Table 6.6 reveals that the performance bound is also violated in this case, with a moderately high probability of 2×10^{-2}. The worst case performance is found to be nearly 10% greater than the allowed value, and Figure 6.22 shows a distinct peak associated with this violation, which occurs between 1.25 and 1.35 Mach. These Mach numbers are expected to be achieved shortly after 25 seconds after lift-off. Figure 6.19 shows that most of the worst case aerodynamic parameters are less than nominal, with the air density close to its minimum value. This is rather counter-intuitive, as the performance is directly proportional to the air density, since $Q = \frac{1}{2}\rho_a V^2$, where ρ_a is the air density. Several parameter values are also close to their nominal values, particularly the MCI parameters, the burn time dTc and the two thruster misalignments. The specific impulse is at the minimum value, as is the first thruster gain $TVC_S F_A$.

The wind model 3 performance violates the safety bound with the maximum violation lying about 18.5% beyond it. Table 6.6 shows that the probability of the performance being greater than 1.15 is low, at 8.2×10^{-21}. However, the probability of the performance being greater than 1 is 1.385×10^{-6}, which is relatively high. It can be seen from Figure 6.22 that the problematic behaviour occurs between 0.3 and 0.4 Mach, i.e.

shortly after lift-off. Observe that the magnitude of $Q\alpha$ is close to 10^5 $Pa.deg$, which is relatively small. However it is experimentally determined that at lower Mach numbers, even such a magnitude can cause structural issues [126]. In fact, this wind model causes the highest violation of bounds in terms of percentage above the safety envelope. The uncertain parameters associated with the worst case performance in the fourth bar plot in Figure 6.19 reveal that the MCI parameters deviate moderately from their nominal values (certainly when compared to the other wind cases). In particular, it is seen that the longitudinal structural COG is nearly at its maximum, while the lateral structural COG is beyond 50% to its minimum. Structural COG deviations indicate elastic mode effects, which could explain the poor performance.

It is interesting to note how the different wind models affect the performance, and how the different uncertain parameters are "activated" by different wind models. One striking observation is the propulsion parameters effectively change direction when wind model 4 is considered. The fourth wind model also appears to occur when the MCI parameters deviate farther from their nominal values, while the aerodynamic coefficients lie close to their nominal values. Note that analyses of how each uncertain parameter impacts the performance value is studied earlier, notably in [129], which inferred that the effect of single parameters do not cause performance bound violations.

Since these wind models are representative of real winds experienced at the launch site, their effects are crucial, and the fact that they cause aerodynamic loads beyond the safety envelope is concerning. One striking observation is that the propulsion parameters change directions. The uncertainty in the specific impulse $dIsp$ and the burn time dTc are both negative (i.e. lower than their nominal values) for the first two wind cases. However, for the wind case 02_1991, dTc reverses direction, and for 07_2005 both dTc and $dIsp$ reverse directions.

Table 6.6: Salient results of the CE-based analyses of the P80, comparing three wind models

	No wind	**Wind model 1: 01_1993**	**Wind model 2: 02_1991**	**Wind model 3: 07_2005**
Smallest non-zero probability	2.52×10^{-23} at γ=0.5	1.206×10^{-24} at γ=1	2×10^{-2} at γ=1	8.2×10^{-21} at γ=1.15
$\gamma\|_{\mathbf{P[J\geq\gamma]}}$	0.55	1.05	1.1	1.2
CE Worst case	0.5263	1.0226	1.0951	1.1841

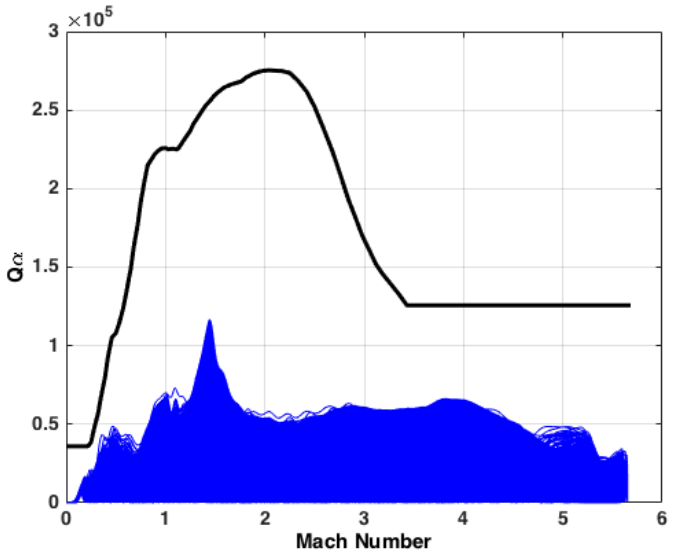


Figure 6.20: $Q\alpha$ versus Mach number trajectories considering no winds in the atmospheric flight phase. The performance does not exceed 55% of the safety bound, and thus robust performance is guaranteed.

It is seen that the presence of any wind model leads to a violation of performance bounds. The results of this study are significant because the Vega launcher is found to satisfy probabilistic robust performance in all cases, despite not satisfying the deterministic robust performance criterion. Unlike in previous industrial models, CE is applied in isolation, without the support of optimisation-based analysis. It is seen that CE performs well in finding the worst case performances, and in assessing the associ-

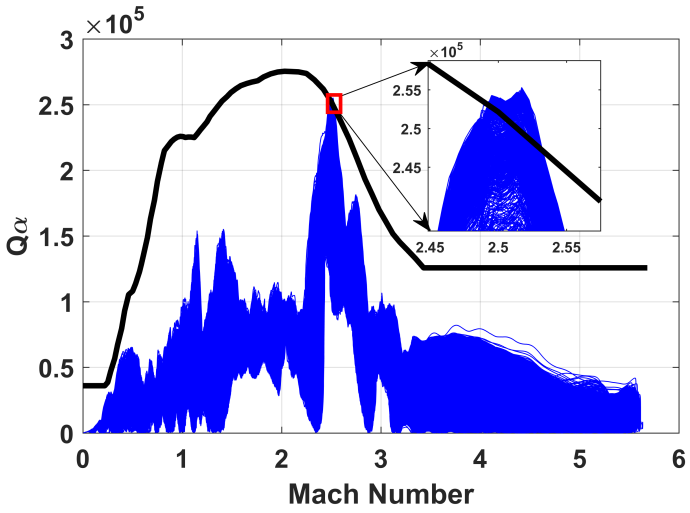


Figure 6.21: $Q\alpha$ versus Mach number trajectories considering wind model 1 in the atmospheric flight phase, showing a violation of the safety envelope close to 2.5 Mach.

ated probabilities.

Some limits to this approach do exist, particularly when applied in the post-flight or post-design phases. The principal limitation, as previously discussed, is that CE cannot consider a large number of uncertain parameters. In the context of VEGACONTROL, an even larger uncertainty set of upto a hundred parameters can be varied. However CE is unable to account for these in its analysis. In addition, the system parameters are considered independent, even if a correlation exists between them. This may result in some worst cases that are not physically feasible. At the current time, the worst cases are discussed and screened for feasibility post analysis, in a meeting with system experts. It may be possible to involve system expert's knowledge before analysis as well in a systematic way (not as currently done, using vertex type methods). This involves expert elicitation, and could be considered for future work using Bayesian type methods.

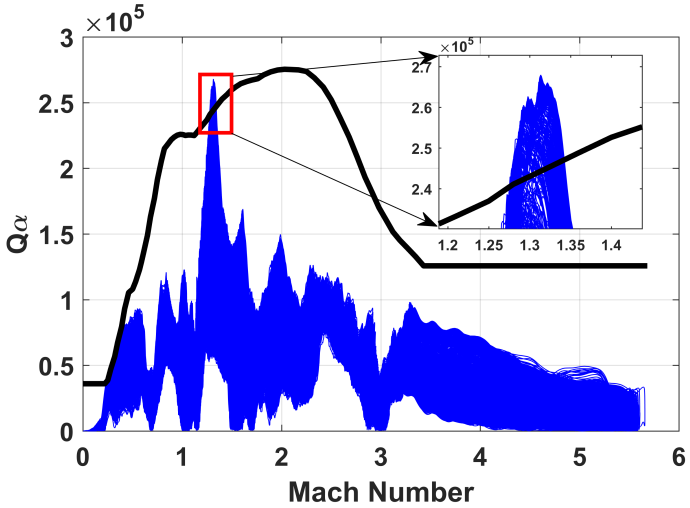


Figure 6.22: $Q\alpha$ versus Mach number trajectories considering wind model 2 in the atmospheric flight phase, showing a significant violation of the safety envelope near 1.3 Mach.

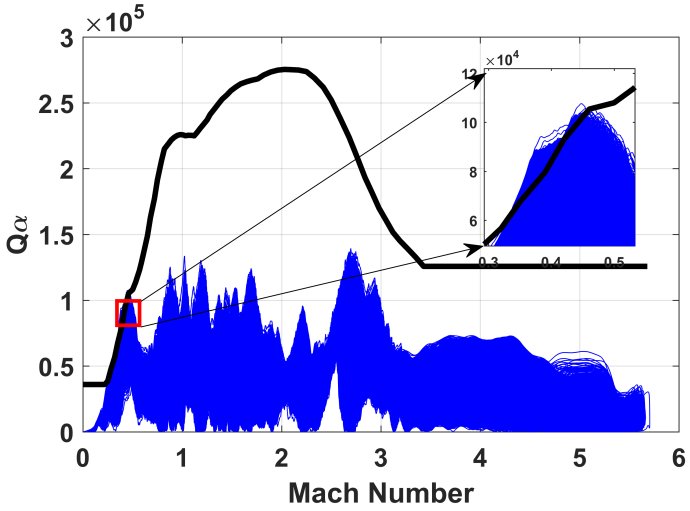


Figure 6.23: $Q\alpha$ versus Mach number trajectories considering wind model 3 in the atmospheric flight phase, showing a violation of the safety envelope close to 0.5 Mach.

6.3 Conclusions

In this chapter, CE is applied to two space industry control systems at different ends of the control synthesis cycle. The rationale for this is to increase the technological maturity of CE as a robust performance assessment tool for space industry models.

CE is first applied to a control system modelling the so-called barbecue mode of a long coasting upper stage of a launcher, where the performance criterion is the total number of thruster actuations. Apart from the modelling uncertainties, configuration parameters related to the upper stage payload are also included in the analysis. The analysis identifies that the worst case found using optimisation is indeed a rare event, with a very low probability of occurrence (of the order 10^{-25}). Although the performance bound is violated, the probability associated with such a violation is also rare, of the order 10^{-8}. In other words, the barbecue mode controller satisfies probabilistic robust performance, but not deterministic robust performance. In this way, CE overcomes the pitfalls of conservatism generally associated with worst case analysis methods.

CE is then applied to a high-fidelity, nonlinear simulator of the Vega launcher model. This implementation is done post-synthesis. The performance criterion is the aerodynamic loads experienced by the P80, Vega's first stage, in atmospheric flight. The structural loads in atmospheric flight is known for being particularly sensitive to wind gust, and a safety envelope for the structural load is generated from experimental data. Two different cases are studied: first, a critical subset consisting of 4 uncertain parameters to investigate their impact on the structural load in the presence of wind gust. The second case is that of implementing various real wind models to assess their effect on the structural loads. In both cases, CE successfully identifies violation of performance, generates the probability profile of performance and computes the probability associated with the worst case.

This chapter demonstrates the capabilities of CE in industry models of two different sizes in different parts of the control synthesis cycle. The following chapter summarises the previous chapters and provides a conclusion to this thesis.

Chapter 7

Conclusions

This thesis proposes a cross entropy simulation-based technique to probabilistically assess the robust performance of spacecraft control systems, and to complement existing robust performance analysis techniques. This research study is motivated by the increasing importance of accurate $V\&V$ in ensuring space mission safety and success, in the light of greater GNC autonomy required in future missions. The most critical stage of $V\&V$ for spacecraft safety is worst case analysis, currently performed in the industry using Monte Carlo or increasingly, using optimisation methods. The worst case analysis of high-fidelity spacecraft dynamics is challenging, because they are described by complex, non-linear, high-order models with a large number of modelling uncertainties.

The analytical robust performance method $\boldsymbol{\mu}$-analysis, rooted in rigorous linear algebraic principles, is attractive in the early design stages in both performance analysis and synthesis. However, it cannot be applied on high-fidelity industry models for three reasons. Firstly, its computational complexity does not scale polynomially with the problem order. Secondly, $\boldsymbol{\mu}$-analysis operates on linear or nearly linear models. Thirdly, the worst case performance reported by $\boldsymbol{\mu}$-analysis is often highly conservative. In addition, modelling high-order industrial systems as LFTs is cumbersome.

The state-of-practice in the space industry is to carry out large-scale classical Monte Carlo simulations, which incur significant computational costs. It is widely accepted that MC is better suited to determining the average performance and the associated standard deviation, rather than extreme performance. Consequently, optimisation-based methods are now being employed for $V\&V$, as they are highly efficient at computing extrema. However, such analyses are also plagued by the issue of conservatism, since the worst case often has a low likelihood of occurrence, and may thus be regarded as a rare-event. Hence, assigning probabilistic measures to the degraded performance is vital. Therefore, this research work aims to bridge the critical gap between the pessimistic worst case performance and the optimistic view of performance often presented by Monte Carlo simulations.

The proposed methodology relies on CE, an adaptive importance sampling technique whose principle is centred in minimising the Kullback-Liebler divergence. CE, well-established in the literature as being effective at estimating rare-event probabilities, is applied in an aerospace performance analysis problem for the first time in this work. Based on CE, a probability profile of performance curve of a closed-loop system, is introduced as a tool for robust performance analysis in this thesis[1]. This tool enables decision makers to determine the expected levels of performance, the probability of bounds violation if any, the worst case performance and the associated probability.

The CE-based PPoP generation algorithm is part of a rigorous performance analysis of the iGNC rendezvous model, alongside $\boldsymbol{\mu}$-analysis and DE-based worst case analysis. Together, these methods compute the worst case performance (all three methods), the deterministic robust performance margins ($\boldsymbol{\mu}$-analysis), and the probability associated with various performance levels, including the worst case performance. When imple-

[1] Although several analogous curves do exist in myriad fields, such as the survival curve, phase transition curve or titration curve.

mented together, these form an integrated analysis template, proposed in this thesis for characterising both deterministic and probabilistic robustness properties of a linear (or linearised) aerospace model.

Implementing CE alongside relatively mature techniques such as μ-analysis and DE also enables verification of its accuracy and reliability on the type of problems of interest in aerospace robust performance analysis. The comparison reveals that the worst case performances arrived at by CE are a close match to those found by the more established techniques. In addition, the worst case uncertain parameters tend towards the same directions in all three cases, implying that the same problematic parameter region is identified by all the methods. These results provide strong evidence that CE may be reliably employed as a robust performance analysis technique, either alone or in conjunction with existing techniques.

CE-based analysis is conducted to assess the robust performance of an attitude controller for a launcher upper phase in the barbecue mode in an early design stage, in which it accounts for payload configuration parameters alongside uncertain parameters. Despite initial oscillation in the means and standard deviations of the parameters, possibly attributable to the integer-value performance function, the analysis successfully identifies a worst case that constitutes a severe violation of its performance bound (this worst case is higher than that found by DE). The strength of the CE-based analysis is that it identifies a very low probability associated with bounds violation, and with the worst case. CE successfully quantifies conservatism, thereby avoiding cumbersome and costly redesign of the controller. With insights from the CE-based analysis, it is determined that the system satisfies probabilistic robustness, even though deterministically speaking, the robust performance is not satisfied.

The Vega launch vehicle's first stage controller is analysed for structural loads violation during atmospheric flight. The CE-based analysis successfully identifies problematic parameter regions. In addition, the probability profile of performance curves are generated considering different wind models. This implementation is significant because the simulator consists of extremely complex non-linear models, with stochastic factors and a moderate number of uncertain parameters (28, to be exact), and so is an excellent benchmark for the complexity of models to be analysed in the space industry. It is noted that CE provides good insight into the performance in both parts of the analysis, identifying where the performance bounds are not robustly satisfied. In addition, CE demands a fraction of the computational time that MC run does, while still identifying poorer performances.

Apart from the CE analysis paradigm, this thesis makes other contributions, which include the development of a set of LFT models of the simplified rendezvous system for MSR, and guaranteeing robust performance of the designed controller. Further, the high-fidelity non-linear iGNC model is analysed using DE to guarantee the designed controller performance in the presence of over 200 uncertainties. Unfortunately, CE is incapable of working with such a large uncertain parameter dimension, and could not be applied.

Taken together, the results presented in this thesis suggest that the CE-based analysis technique is an excellent candidate for determining probabilistic robustness in aerospace systems. The most important strength of this technique is its ability to determine probabilistic robustness of a system, and to quantify conservatism of worst case performance. Further, the CE-based method is attractive in the industrial $V\&V$ context because it is applicable in existing $V\&V$ frameworks — in terms of the inputs and outputs, its behaviour appears reasonably similar to classical Monte Carlo, and analysts do not

need to apply engineering effort before each analysis. In most cases, the only engineering decisions needed before a CE analysis are the number of samples allowed in each iteration, and the total number of CE iterations allowed. These values depend on the time taken for a single evaluation of the model, and the computational budget available.

The most important limitation of the CE-based technique developed in this study is that the uncertain parameter it can handle is (relatively) modest, due to the effect of likelihood ratio degeneracy. In the process of implementing the technique, the most number of parameters that was successful was in the dozens, rather than in the hundreds (as demanded by the iGNC nonlinear model, for example). Another limitation is that CE implementation is not readily applicable in the case of parameter distributions not belonging to the exponential distribution family. In the current work, the models required truncated Gaussian distributions, and hence it is convenient. For cases where a uniform distribution is more appropriate, for example, the Kullback-Liebler divergence based updating law must be modified. This will be explored in future work.

The work presented in this thesis has direct application in flight clearance of aerospace control systems. $V\&V$ is the most crucial step of the control design cycle, and is also the most expensive. CE can efficiently determine whether performance criteria are met, and if they are not, it can determine whether costly retuning or redesign of the controller is justified. The integrated analysis template proposed in the thesis is rigorous, yet does not make highly pessimistic recommendations, simply by virtue of the conservatism being quantified. Such an approach to $V\&V$ has the potential to truly balance the crucial aspects of flight safety with the speed demanded by commercial operations to be efficient, and ultimately, profitable.

7.1 Future Work

Future research directions created by the research in this thesis are now identified.
The CE-based analysis technique presented may be implemented to determine the so-called safe and unsafe parameter regions. Although the scattered heat maps presented in the thesis indirectly allude to these concepts, a systematic treatment and mathematical formalisation may be beneficial. The advantage of CE over MC in such applications, is that CE samples more densely from the unsafe parameter regions, thus providing richer data and clearer boundaries.

Further research work is needed to fully develop the CE-based technique into a powerful, self-contained probabilistic analysis technique that can be readily implemented by the space industry. Importantly, it is important to increase the number of uncertain parameters the technique can reliably consider. In order to overcome its limiting cause, i.e., likelihood ratio degeneracy, it may be possible to use principles of optimisation, or Gaussian processes while updating the parameter distributions. It is believed that combining these powerful approaches can also enable CE to improve its computation efficiency even further. Another improvement with regard to computational efficiency that can be implemented more readily, is in parallelising function evaluations, both during CE updating and during the final IS run.

The implementation of CE techniques in control design may be investigated. Information of robust performance being satisfied incrementally from the initial design stage to the final high-fidelity model can be considered right from the beginning of the analysis. This constitutes randomised control design, which is a research area that CE can certainly contribute to. Inspecting the PPoP curve for satisfactory performance in every stage of the design cycle can provide direction for tuning or control.

In view of longer term perspectives, the probabilistic approach to $V\&V$ can be greatly expanded. Expert elicitation techniques may be used to intelligently determine effective prior distributions. For example, the physics of a problem may dictate that a parameter is likely to cause poor performance due to being associated with a singularity in the dynamics. The prior mean parameter can then be biased towards the singularity point, to truly investigate degraded performances. In this way, the human involvement in $V\&V$ may occur at higher levels of cognition, and much of the human engineering effort currently being conducted may be phased out. Such levels of autonomy can facilitate faster, more efficient $V\&V$ in lesser time, and can ultimately greatly enhance space system safety and performance.

References

[1] Tosney, W. F. and Cheng, P. G., "Space Safety is no Accident how the Aerospace Corporation Promotes Space Safety," *Space Safety is No Accident*, Springer, 2015, pp. 101–108. 1

[2] Croomes, S., "Overview of the DART mishap investigation results," *NASA Report, available online at nasa.gov/pdf/148072main_DART_mishap_overview.pdf*, 2006, pp. 1–10. 1

[3] SpaceX Inc., "SpaceX CRS-7 Mission Press Kit," *Available online at spacex.com/sites/spacex/files/spacex_nasa_crs-7_presskit*, June 2015. 1

[4] SpaceX Inc., "CRS-7 Investigation Update," *Available online at spacex.com/news/2015/07/20/crs-7-investigation-update*, July 2015. 1

[5] Wong, W. E., Debroy, V., Surampudi, A., Kim, H., and Siok, M. F., "Recent catastrophic accidents: Investigating how software was responsible," *Secure Software Integration and Reliability Improvement (SSIRI), 2010 Fourth International Conference on*, IEEE, 2010, pp. 14–22. 2

[6] Tafazoli, M., "A study of on-orbit spacecraft failures," *Acta Astronautica*, Vol. 64, No. 2, 2009, pp. 195–205. 2

[7] Baglioni, P., Fisackerly, R., Gardini, B., Gianfiglio, G., Pradier, A. L., Santovincenzo, A., Vago, J. L., and van Winnendael, M., "The Mars exploration plans of

ESA," *IEEE Robotics & Automation Magazine*, Vol. 13, No. 2, 2006, pp. 83–89. 2

[8] Zolghadri, A., "Advanced model-based FDIR techniques for aerospace systems: Today challenges and opportunities," *Progress in Aerospace Sciences*, Vol. 53, 2012, pp. 18–29. 3

[9] Fehse, W., *Automated rendezvous and docking of spacecraft*, Vol. 16, Cambridge university press, 2003. 3, 13, 81

[10] Philippe, C., "Verification, validation, and certification of aerospace control systems," *The Impact of Control Technology*, February 2011, pp. 205 – 207. 3

[11] Wang, W., *Worst-case Analysis of Space Systems*, Ph.D. thesis, University of Exeter, 2011. 3, 11, 25, 32, 39, 125, 161, 169

[12] Doyle, J., Packard, A., and Zhou, K., "Review of LFTs, LMIs, and μ," *Decision and Control, 1991., Proceedings of the 30th IEEE Conference on*, IEEE, 1991, pp. 1227–1232. 3, 23

[13] Rubinstein, R. and Kroese, D., *Simulation and the Monte Carlo method*, John Wiley & Sons, 2011. 3, 53, 55, 59, 160

[14] Fielding, C., Varga, A., Bennani, S., and Selier, M., *Advanced Techniques for Clearance of Flight Control Laws*, Lecture Notes in Control and Information Sciences, Springer Berlin Heidelberg, 2002. 4, 10, 32, 38

[15] Skogestad, S. and Postlethwaite, I., *Multivariable feedback control: analysis and design*, Vol. 2, Wiley New York, 2007. 9, 14, 15, 16, 17, 18, 19, 22, 55, 104, 123

[16] Oberkampf, W. L., Trucano, T. G., and Hirsch, C., "Verification, validation, and predictive capability in computational engineering and physics," *Applied Mechanics Reviews*, Vol. 57, No. 5, 2004, pp. 345–384. 10, 13

[17] Jacklin, S. A., Lowry, M. R., Schumann, J. M., Gupta, P. P., Bosworth, J. T., Zavala, E., Kelly, J. W., Hayhurst, K. J., Belcastro, C. M., and Belcastro, C. M., "Verification, validation, and certification challenges for adaptive flight-critical control system software," *Proceedings of AIAA Guidance Navigation and Control Conference and Exhibit*, 2004. 10

[18] Kamath, A. K., *Surrogate-Assisted Optimisation-Based Verification & Validation*, Ph.D. thesis, 2014. 11, 36, 40, 41, 45, 77, 125, 146

[19] Mankins, J. C., "Technology readiness assessments: A retrospective," *Acta Astronautica*, Vol. 65, No. 9, 2009, pp. 1216–1223. 12

[20] Joos, H.-D., "A methodology for multi-objective design assessment and flight control synthesis tuning," *Aerospace Science and Technology*, Vol. 3, No. 3, 1999, pp. 161–176. 12

[21] Korte, U., "Tasks and needs of the industrial clearance process," *Advanced Techniques for Clearance of Flight Control Laws*, Springer, 2002, pp. 13–33. 12

[22] Korte, U. and Selier, M., "Introduction," *Advanced Techniques for Clearance of Flight Control Laws*, Springer, 2002, pp. 13–33. 13

[23] Safonov, M. G., "Origins of robust control: Early history and future speculations," *Annual Reviews in Control*, Vol. 36, No. 2, 2012, pp. 173–181. 14, 21

[24] Zhou, K. and Doyle, J. C., *Essentials of Robust Control*, Prentice Hall, Upper Saddle River, NJ, 1998. 14, 15, 16, 21, 26, 123

[25] O'Hagan, A. and Oakley, J. E., "Probability is perfect, but we can't elicit it perfectly," *Reliability Engineering & System Safety*, Vol. 85, No. 1-3, jul 2004, pp. 239–248. 15

[26] Hanson, J. and Beard, B., "Applying Monte Carlo simulation to launch vehicle design and requirements analysis," 2010. 15, 16, 37, 63, 64

[27] Agarwal, H., Renaud, J. E., Preston, E. L., and Padmanabhan, D., "Uncertainty quantification using evidence theory in multidisciplinary design optimization," *Reliability Engineering & System Safety*, Vol. 85, No. 1, 2004, pp. 281–294. 16

[28] Sutcliffe, A. and Sawyer, P., "Requirements elicitation: Towards the unknown unknowns," *2013 21st IEEE International Requirements Engineering Conference (RE)*, IEEE, 2013, pp. 92–104. 16, 35

[29] Packard, A. and Doyle, J., "The complex structured singular value," *Automatica*, Vol. 29, No. 1, 1993, pp. 71–109. 16, 25, 27, 28

[30] Doyle, J., "Structured uncertainty in control system design," *1985 24th IEEE Conference on Decision and Control*, No. 24, 1985, pp. 260–265. 17, 24

[31] Doyle, J., "Analysis of feedback systems with structured uncertainties," *IEE Proceedings D-Control Theory and Applications*, Vol. 129, IET, 1982, pp. 242–250. 17, 26, 27

[32] Lagoa, C. M. and Barmish, B. R., "Distributionally robust Monte Carlo simulation: a tutorial survey," Vol. 35, Elsevier, 2002, pp. 151–162. 17, 45, 50

[33] Hecker, S. and Varga, A., "Generalized LFT-based representation of parametric uncertain models," *European Control Conference (ECC), 2003*, IEEE, 2003, pp. 763–768. 17

[34] Bates, D. G., "H_∞ control design," *Mathematical Methods for Robust and Nonlinear Control*, 2007, pp. 3–45. 18

[35] Zhou, K., Doyle, J. C., and Glover, K., *Robust and optimal control*, Vol. 40, Prentice hall New Jersey, 1996. 18, 19

[36] Veenman, J., Koroglu, H., and Scherer, C., "Analysis of the controlled NASA HL20 atmospheric re-entry vehicle based on dynamic IQCs," *AIAA Guidance, Navigation and Control Conference*, 2009. 19, 29

[37] Bates, D. and Postlethwaite, I., *Robust multivariable control of aerospace systems*, Vol. 8, IOS Press, 2002. 20, 28

[38] Balas, G. J., "Flight control law design: An industry perspective," *European Journal of Control*, Vol. 9, No. 2, 2003, pp. 207–226. 20

[39] Doyle, J. and Stein, G., "Multivariable Feedback Design: Concepts for a Classical/Modern Synthesis," *IEEE Transactions on Automatic Control*, , No. 1, 1981, pp. 4–16. 21

[40] Marcos, A., Bates, D. G., and Postlethwaite, I., "Nonlinear symbolic LFT tools for modelling, analysis and design," *Nonlinear Analysis and Synthesis Techniques for Aircraft Control*, Springer, 2007, pp. 69–92. 21

[41] Hecker, S. and Varga, A., "Symbolic manipulation techniques for low order LFT-based parametric uncertainty modelling," *International Journal of Control*, Vol. 79, No. 11, 2006, pp. 1485–1494. 25

[42] Beck, C. L., Doyle, J., and Glover, K., "Model reduction of multidimensional and uncertain systems," *Automatic Control, IEEE Transactions on*, Vol. 41, No. 10, 1996, pp. 1466–1477. 25

[43] Braatz, R. D. and Crisalle, O. D., "Robustness analysis for systems with ellipsoidal uncertainty," *International journal of robust and nonlinear control*, Vol. 8, No. 13, 1998, pp. 1113–1117. 25

[44] Dahleh, M., Dahleh, M. A., and Verghese, G., "Stability Robustness," *Lectures on dynamic systems and control, MIT*, 2011. 26

[45] Roos, C. and Biannic, J.-M., "A detailed comparative analysis of all practical algorithms to compute lower bounds on the structured singular value," *Control Engineering Practice*, Vol. 44, 2015, pp. 219–230. 27, 28

[46] Fan, M. K. and Tits, A. L., "Characterization and efficient computation of the structured singular value," *Automatic Control, IEEE Transactions on*, Vol. 31, No. 8, 1986, pp. 734–743. 27

[47] Braatz, R. P., Young, P. M., Doyle, J. C., and Morari, M., "Computational complexity of μ calculation," *Automatic Control, IEEE Transactions on*, Vol. 39, No. 5, 1994, pp. 1000–1002. 28

[48] Fabrizi, A., Roos, C., and Biannic, J.-M., "A detailed comparative analysis of μ lower bound algorithms," *European Control Conference 2014*, 2014. 28

[49] Fan, M. K., Tits, A. L., and Doyle, J. C., "Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics," *IEEE Transactions on Automatic Control*, Vol. 36, No. 1, 1991, pp. 25–38. 28

[50] Safonov, M. G., "Origins of robust control: Early history and future speculations," *Annual Reviews in Control*, Vol. 36, No. 2, 2012, pp. 173–181. 29, 135, 136

[51] Bernstein, D. S., Haddad, W. M., and Sparks, A. G., "A Popov criterion for uncertain linear multivariable systems," *Automatica*, Vol. 31, No. 7, 1995, pp. 1061–1064. 29

[52] Anderson, J. and Papachristodoulou, A., "Robust nonlinear stability and performance analysis of an F/A-18 aircraft model using sum of squares programming," *International Journal of Robust and Nonlinear Control*, Vol. 23, No. 10, 2013, pp. 1099–1114. 29, 30

[53] Roos, C., Biannic, J.-M., Tarbouriech, S., and Prieur, C., "On-ground aircraft control design using an LPV anti-windup approach," 2007, pp. 117–145. 29

[54] Metropolis, N. and Ulam, S., "The Monte Carlo method," *Journal of the American statistical association*, Vol. 44, No. 247, 1949, pp. 335–341. 31

[55] Roux, C., Mantini, V., Marcos, A., Peñín, L. F., and Bennani, S., "Robust Flight Control System Design Verification & Validation for Launchers," *AIAA Guidance, Navigation, and Control Conference*, 2012, p. 4996. 32

[56] Silva, S., Lopes Junior, V., and Assunção, E., "Robust control to parametric uncertainties in smart structures using linear matrix inequalities," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. 26, No. 4, 2004, pp. 430–437. 32, 127

[57] Kroese, D., Taimre, T., and Botev, Z., *Handbook of Monte Carlo Methods*, John Wiley & Sons, Inc., Hoboken, NJ, USA, feb 2011. 34, 56, 63, 69

[58] Tempo, R., Calafiore, G., and Dabbene, F., *Randomized algorithms for analysis and control of uncertain systems: with applications*, Springer Science & Business Media, 2012. 34, 36, 46, 49

[59] Bastien, P., Martin, S., Vincent, R., and El-Baz, D., "Monte-Carlo analysis of object reentry in Earth's atmosphere based on Taguchi method," *8th European Symposium on Aerothermodynamics for Space Vehicles*, 2015. 34, 35

[60] Gu, J., Zhang, S., and Wang, B., "Monte Carlo Analysis for Significant Parameters Ranking in RLV Flight Evaluation," *Procedia Engineering*, Vol. 99, 2015, pp. 1082–1088. 34

[61] Taguchi, G. and Taguchi, G., "System of experimental design; engineering methods to optimize quality and minimize costs," Tech. rep., 1987. 35

[62] Baumann, E., Bahm, C., Strovers, B., Beck, R., and Richard, M., "The X-43A six degree of freedom Monte Carlo analysis," *AIAA Aerospace Sciences Meeting and Exhibit, Paper No*, Vol. 203, 2008, p. 2008. 35

[63] Desai, P. N., Braun, R. D., Powell, R. W., Engelund, W. C., and Tartabini, P. V., "Six-degree-of-freedom entry dispersion analysis for the METEOR recovery module," *Journal of Spacecraft and Rockets*, Vol. 34, No. 3, 1997, pp. 334–340. 35

[64] Way, D. W., Powel, R. W., Chen, A., Steltzner, A. D., Burkhart, P. D., and Mendeck, G. F., "Mars Science Laboratory: Entry, descent, and landing system performance," *Aerospace Conference, 2007 IEEE*, IEEE, 2007, pp. 1–19. 35

[65] Olsson, A., Sandberg, G., and Dahlblom, O., "On Latin hypercube sampling for structural reliability analysis," *Structural safety*, Vol. 25, No. 1, 2003, pp. 47–68. 36

[66] Rubino, G. and Tuffin, B., *Rare event simulation using Monte Carlo methods*, John Wiley & Sons, 2009. 37, 46, 53, 148

[67] Ravindran, A., Reklaitis, G. V., and Ragsdell, K. M., *Engineering optimization: methods and applications*, John Wiley & Sons, 2006. 38

[68] Wang, W., Menon, P., Bates, D., Ciabuschi, S., Gomes Paulino, N. M., Di Sotto, E., Bidaux, A., Kron, A., Salehi, S., and Bennani, S., "Verification and Validation Framework for Autonomous Rendezvous Systems in Terminal Phase," *Journal of Spacecraft and Rockets*, Vol. 52, No. 2, 2014, pp. 625–629. 38, 41, 125, 161

[69] Menon, P. P., Bates, D. G., and Postlethwaite, I., "Nonlinear robustness analysis of flight control laws for highly augmented aircraft," *Control Engineering Practice*, Vol. 15, No. 6, 2007, pp. 655–662. 38, 40

[70] Menon, P., Postlethwaite, I., Bennani, S., Marcos, A., and Bates, D. G., "Robustness analysis of a reusable launch vehicle flight control law," *Control Engineering Practice*, Vol. 17, No. 7, 2009, pp. 751–765. 38

[71] Joos, H.-D., "Application of Optimization-Based Worst Case Analysis to Control Law Assessment in Aerospace," *Advances in Aerospace Guidance, Navigation and Control*, Springer, 2015, pp. 53–65. 38

[72] Menon, P. P., Kim, J., Bates, D. G., and Postlethwaite, I., "Clearance of nonlinear flight control laws using hybrid evolutionary optimization," *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 6, 2006, pp. 689–699. 40, 125

[73] Granville, V., Krivánek, M., and Rasson, J.-P., "Simulated annealing: A proof of convergence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 6, 1994, pp. 652–656. 40

[74] Bennani, S., Wang, W., Menon, P., and Bates, D., "Robustness analysis of attitude and orbit control systems for flexible satellites," *IET Control Theory & Applications*, Vol. 4, No. 12, dec 2010, pp. 2958–2970. 41, 58

[75] Paulino, N., Di Sotto, E., Salehi, S., Kron, A., Hamel, J.-F., Wang, W., Menon, P., Bates, D., Papachristodoulou, A., and Maier, C., "VVAF–Worst case & safety analysis tools for autonomous rendezvous system," *AIAA Guidance, Navigation, and Control Conference*, 2010, p. 7904. 41, 125

[76] Storn, R. and Price, K., "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization*, Vol. 11, No. 4, 1997, pp. 341–359. 41, 42

[77] Das, S. and Suganthan, P. N., "Differential evolution: a survey of the state-of-the-art," *Evolutionary Computation, IEEE Transactions on*, Vol. 15, No. 1, 2011, pp. 4–31. 41

[78] Calafiore, G., Dabbene, F., and Tempo, R., "Randomized algorithms for probabilistic robustness with real and complex structured uncertainty," *IEEE Transactions on Automatic Control*, Vol. 45, No. 12, 2000, pp. 2218–2235. 44

[79] Chen, X., Zhou, K., and Aravena, J., "Probabilistic robustness analysis-risks, complexity, and algorithms," *SIAM Journal on Control and Optimization*, Vol. 47, No. 5, 2008, pp. 2693–2723. 45

[80] Stengel, R. F., "Some effects of parameter variations on the lateral-directional stability of aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 3, No. 2, 1980, pp. 124–131. 45

[81] Tempo, R., Bai, E.-W., and Dabbene, F., "Probabilistic robustness analysis: Explicit bounds for the minimum number of samples," *Decision and Control, 1996., Proceedings of the 35th IEEE Conference on*, Vol. 3, IEEE, 1996, pp. 3424–3428. 45, 46

[82] Khargonekar, P. and Tikku, A., "Randomized algorithms for robust control analysis and synthesis have polynomial complexity," *Decision and Control, 1996., Proceedings of the 35th IEEE Conference on*, Vol. 3, IEEE, 1996, pp. 3470–3475. 45

[83] Lagoa, C. M., Shcherbakov, P. S., and Barmish, B. R., "Probabilistic enhancement of classical robustness margins: the unirectangularity concept," *Systems & Control Letters*, Vol. 35, No. 1, 1998, pp. 31–43. 45

[84] Khatri, S. and Parrilo, P., "Guaranteed bounds for probabilisticμ," *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, Vol. 3, IEEE, 1998, pp. 3349–3354. 45, 49

[85] Balas, G. J., Seiler, P. J., and Packard, A., "Analysis of an UAV Flight Control System Using Probabilistic μ," *Proceedings of the AIAA Guidance, Navigation*

and Control Conference, Minneapolis, Minnesota, August, 2012, pp. 13–16. 49, 50

[86] Zhu, X., "Improved bounds computation for probabilistic μ," *American Control Conference, 2000. Proceedings of the 2000*, Vol. 6, IEEE, 2000, pp. 4336–4340. 50

[87] Halder, A., Lee, K., and Bhattacharya, R., "A Probabilistic Method for Nonlinear Robustness Analysis of F-16 Controllers," *arXiv preprint arXiv:1402.0147*, 2014. 50

[88] Nagy, Z. K. and Braatz, R. D., "Worst-case and distributional robustness analysis of finite-time control trajectories for nonlinear distributed parameter systems," *Control Systems Technology, IEEE Transactions on*, Vol. 11, No. 5, 2003, pp. 694–704. 50

[89] Lee, K., Halder, A., and Bhattacharya, R., "Performance and robustness analysis of stochastic jump linear systems using Wasserstein metric," *Automatica*, Vol. 51, 2015, pp. 341–347. 50

[90] Rubinstein, R., "Optimization of computer simulation models with rare events," *European Journal of Operational Research*, Vol. 99, No. 96, 1997, pp. 89–112. 52, 63

[91] Botev, Z. I., Kroese, D. P., Rubinstein, R. Y., and LEcuyer, P., "The cross-entropy method for optimization," *Machine Learning: Theory and Applications*, Vol. 31, 2013, pp. 35–59. 53

[92] Kroese, D., Porotsky, S., and Rubinstein, R., "The Cross-Entropy Method for Continuous Multi-Extremal Optimization," *Methodology and Computing in Applied Probability*, Vol. 8, No. 3, Oct. 2006, pp. 383–407. 53, 62, 63, 64, 148, 172

[93] de Boer, P., Kroese, D., Mannor, S., and Rubinstein, R., "A Tutorial on the Cross-Entropy Method," *Annals of Operations Research*, Vol. 134, No. 1, feb 2005, pp. 19–67. 53

[94] Hui, K., "The Cross-Entropy Method for Network Reliability," *Annals of Operations Research*, 2005, pp. 101–118. 53, 148

[95] Kobilarov, M., "Cross-entropy motion planning," *The International Journal of Robotics Research*, Vol. 31, No. 7, 2012, pp. 855–871. 53

[96] González-Fernández, R. and Leite Da Silva, A., "Reliability assessment of time-dependent systems via sequential cross-entropy Monte Carlo simulation," *IEEE Transactions on Power Systems*, Vol. 26, No. 4, 2011, pp. 2381–2389. 53

[97] Kullback, S. and Leibler, R. A., "On information and sufficiency," *The annals of mathematical statistics*, Vol. 22, No. 1, 1951, pp. 79–86. 53

[98] Montgomery, D. and Runger, G., *Applied Statistics and Probability for Engineers*, John Wiley & Sons, 2011. 53, 54, 56

[99] Morio, J. and Balesdent, M., *Estimation of Rare Event Probabilities in Complex Aerospace and Other Systems: A Practical Approach*, Woodhead Publishing, 2015. 53, 187

[100] Taylor, H. M. and Karlin, S., *An introduction to stochastic modeling*, Academic press, 2014. 53, 157

[101] Diaconis, P. and Mosteller, F., "Methods for studying coincidences," *Journal of the American Statistical Association*, Vol. 84, No. 408, 1989, pp. 853–861. 56

[102] Menon, P., Postlethwaite, I., Bennani, S., Marcos, A., and Bates, D., "Robustness analysis of a reusable launch vehicle flight control law," *Control Engineering Practice*, Vol. 17, No. 7, jul 2009, pp. 751–765. 58

[103] Wang, W., Menon, P., Bates, D., Ciabushi, S., Paulino, N., Bidaux, A., Garus, A., Kron, A., Salehi, S., and Bennani, S., "Verification and validation of autonomous rendezvous systems in the terminal phase," *AIAA Guidance Navigation and Control*, 2012, pp. 1–11. 58

[104] Mujumdar, A., Menon, P. P., Edwards, C., and Bennani, S., "Characterising the probability profile of performance degradation of a Darwin satellite controller," *Control Conference (ECC), 2014 European*, IEEE, 2014, pp. 2514–2519. 61

[105] Calafiore, G., Dabbene, F., and Tempo, R., "Research on probabilistic methods for control system design," *Automatica*, Vol. 47, No. 7, july 2011, pp. 1279–1293. 73, 74

[106] Strippoli, L. and Colmenenarejo, P., "Rendezvous scenario consolidation: IGNC," Tech. rep., GMV, Madrid, 01 2011, GMV-AIR-GNC-TN1.3. 78

[107] iMARS Working Group, "Preliminary planning for an international Mars sample return mission: report of the International Mars Architecture for the Return of Samples (iMARS) working group, white paper, posted July 2008 by the Mars Exploration Program Analysis Group (MEPAG)," 2008. 79

[108] Gargaud, M., *Encyclopedia of astrobiology*, Vol. 1, Springer Science & Business Media, 2011. 79

[109] Strippoli, L., Peuvdic, C. L., Rowel, N., and Colmenenarejo, P., "Detailed models: IGNC," Tech. rep., GMV, Madrid, 01 2012, GMV-AIR-GNC-TN2.2. 80

[110] Ankersen, F., *Guidance, Navigation, Control and Relative Dynamics for Spacecraft Proximity Maneuvers*, Ph.D. thesis, Aalborg University, 2011. 81, 95

[111] Wie, B., *Space vehicle dynamics and control*, AIAA, 1998. 81, 92, 93

[112] Manceaux-Cumer, C. and Chrétien, J.-P., "Minimal LFT form of a spacecraft built up from two bodies," *AIAA Guidance, Navigation, and Control Conference and Exhibit, Montreal, Canada*, 2001. 84, 100

[113] Yang, Y., Liu, Z.-W., Shi, W.-X., and Huang, X.-F., "Accurate Measurement of Nonlinear Liquid Sloshing," *AIAA Journal*, Vol. 53, No. 3, 2015, pp. 771–779. 93

[114] Bayle, O., LHullier, V., Ganet, M., Delpy, P., Francart, J.-L., and Paris, D., "Influence of the ATV propellant sloshing on the GNC performance," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Monterey, CA*, 2002. 93

[115] Colmenarejo, P., Gandía, F., García Casas, A., Tomassini, A., and Ankersen, F., "Gncde: An Integrated GNC Development Environment," *6th International ESA Conference on Guidance, Navigation and Control Systems*, Vol. 606, 2006, p. 29. 110, 111

[116] Brat, G., Drusinsky, D., Giannakopoulou, D., Goldberg, A., Havelund, K., Lowry, M., Pasareanu, C., Venet, A., Visser, W., and Washington, R., "Experimental evaluation of verification and validation tools on martian rover software," *Formal Methods in System Design*, Vol. 25, No. 2-3, 2004, pp. 167–198. 117

[117] Bourgault, F., *Model uncertainty and performance analysis for precision controlled space structures*, Master's thesis, Massachusetts Institute of Technology, 2000. 127

[118] MathWorks, I., *Curve fitting toolbox: for use with MATLAB® : user's guide*, MathWorks, 2002. 132

[119] Paulino, N. M., Mujumdar, A., Strippoli, L., Menon, P., and Colmenenarejo, P., "Worst case analysis for GNC design: IGNC," Tech. rep., GMV, Madrid, 01 2014, GMV-AIR-GNC-TN4.7. 171

[120] Strauch, H., Luig, K., and Bennani, S., "Model Based Design Environment for Launcher Upper Stage GNC Development," *Simulation and EGSE for Space Programmes*, ESA, 2015. 175, 176

[121] Strauch, H., Hayward, K., Orlando, G., Sopasakis, P., Mujumdar, A., and Jetzschmann, C., "Design justification: long coasting barbecue phase, sloshing parameter identification and MCI identification," Tech. rep., Airbus Defence and Space, Bremen, 01 2015, USACDF-TN0013. 176, 180, 185

[122] Hughes, P. C., *Spacecraft attitude dynamics*, Courier Corporation, 2012. 179

[123] Snell, A. S., "Preliminary assessment of the robustness of dynamic inversion based flight control laws," *Guidance, Navigation and Control Conference*, AIAA, 1992. 180

[124] Mujumdar, A., Menon, P. P., Roux, C., and Bennani, S., "Cross-Entropy Based Probabilistic Analysis of VEGA Launcher Performance," *Advances in Aerospace Guidance, Navigation and Control*, Springer, 2015, pp. 719–737. 193

[125] Simplício, P., Bennani, S., Marcos, A., Roux, C., and Lefort, X., "Structured Singular-Value Analysis of the Vega Launcher in Atmospheric Flight," *Journal of Guidance, Control, and Dynamics*, 2016, pp. 1342–1355. 195, 197

[126] Valli, M., Spallotta, D., Roux, C., Marcos, A., and Mujumdar, A., "Thrust Vector Control validation results for performance and stability robustness assessments," *EuroGNC 2015, Toulouse, France*, 2015. 196, 197, 199, 211

[127] Marcos, A., De Marina, H. G., Mantini, V., Roux, C., and Bennani, S., "Optimization-based worst-case analysis of a launcher during the atmospheric ascent phase," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2013. 197

[128] Marcos, A., Bennani, S., Roux, C., and Valli, M., "LPV modeling and LFT Uncertainty Identification for Robust Analysis: application to the VEGA Launcher during Atmospheric Phase," *IFAC-PapersOnLine*, Vol. 48, No. 26, 2015, pp. 115–120. 197

[129] Marcos, A., Mantini, V., Roux, C., and Bennani, S., "Bridging the gap between linear and nonlinear worst-case analysis: An application case to the atmospheric phase of the vega launcher," *IFAC Proceedings Volumes*, Vol. 46, No. 19, 2013, pp. 42–47. 197, 211

[130] Bernard, M., *Vega missionization and post flight analyses*, Ph.D. thesis, Sapienza Universita Di Roma, 2009. 202, 206

Appendix A

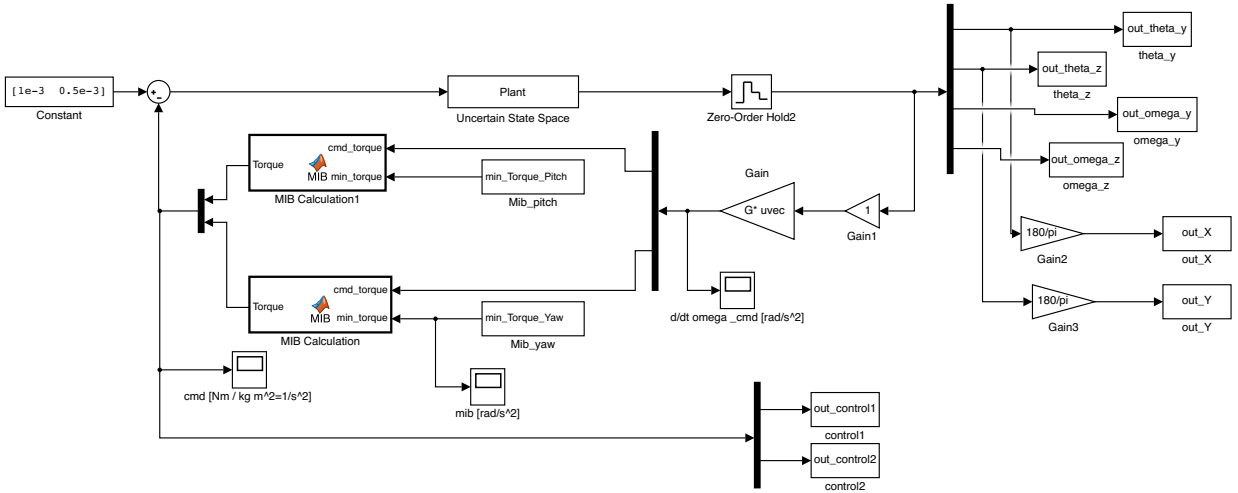


Figure A.1: Simulink interconnection showing the closed loop attitude control system of the long coasting upper stage

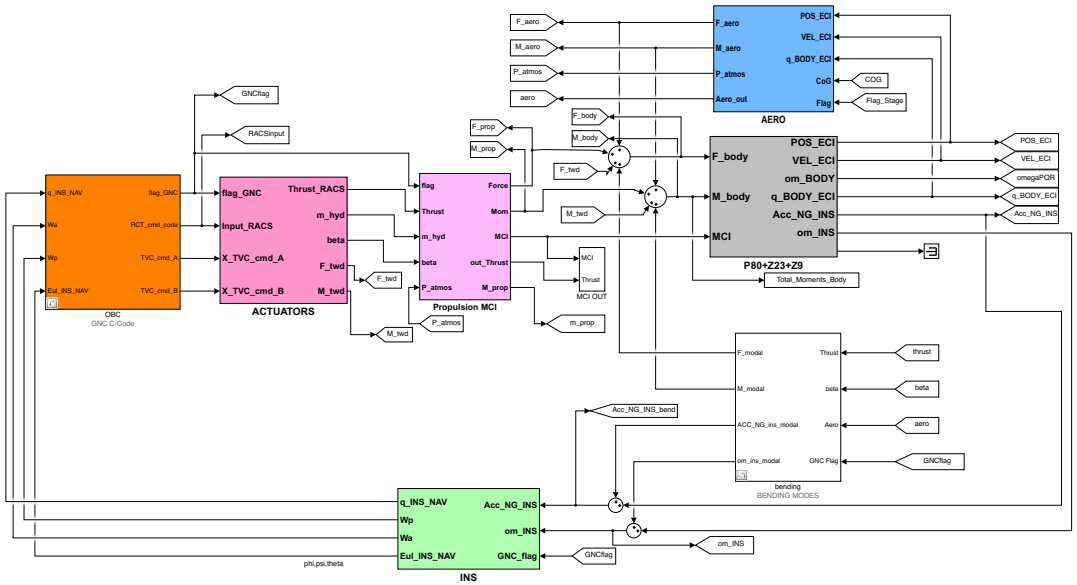


Figure A.2: Simulink interconnection of the Vega launch vehicle P80 dynamics